2003-06

Design and simulation of a three-axis stabilized satellite and Kalman filter rate estimator

Vitalich, John

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DESIGN AND SIMULATION OF A THREE-AXIS STABILIZED SATELLITE AND KALMAN FILTER RATE ESTIMATOR

by

John Vitalich

June 2003

Thesis Advisor: Hal Titus

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 2003	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE
Design And Simulation Of A Three-Axis Stabilized Satellite And Kalman Filter Rate Estimator

5. FUNDING NUMBERS

6. AUTHOR(S) John Vitalich

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)* Design requirements for a small satellite (NPSAT-1) Attitude Determination and Control Subsystem (ADCS) is a three-axis stabilized spacecraft which requires a control attitude of +/- 1.0 degrees and knowledge attitude of +/- 0.1 degree. Several design aspects are considered in development of attitude control systems for a small satellite, such as: spacecraft dynamics, space environment, disturbance torques, orbit type, and spacecraft complexity. The ideal spacecraft's attitude sensor is a rate gyroscope, which provides rate information to the attitude control system. In the case of NPSAT-1, due to budget constraints alternative sensors will be utilized, such as: a three-axis magnetometer, earth sensors, and a Global Positioning System (GPS). A small satellite designed to have a three-axis stabilized, biased momentum system, must have a robust control system and requires a momentum wheel to provide stiffness to maintain attitude, and magnetic torque rods on each axis. The current design of NPSAT-1 uses all of these sensors to provide rate information for damping and stability to the control system that requires a complicated attitude control design. The purpose of this attitude control design simulation is to investigate and propose a control law utilizing a single pitch momentum wheel and three magnetic torque rods. A further proposal is to utilize a constant speed momentum wheel to avoid momentum damping and over speed, replace the pitch control with magnetic torquers, and develop a Kalman filter estimator to provide all the required angular rates.

14. SUBJECT TERMS: Kalman filter, Molniya, MATLAB, SIMULINK, Three axis stabilization, Spacecraft, Satellite, Star Sensor, Estimation, Gyroscope, Rate			15. NUMBER OF PAGES 16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-1

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DESIGN AND SIMULATION OF A THREE-AXIS STABILIZED SATELLITE AND KALMAN FILTER RATE ESTIMATOR

John Vitalich
Lieutenant Commander, United States Navy
B.S., Chapman College, 1989

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

June 2003

Author:

John J. Vitalich

Approved by:

Hal A. Titus
Thesis Advisor

Brij N. Agrawal
Co-Advisor

Jeffrey B. Knorr
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Design requirements for a small satellite (NPSAT-1) Attitude Determination and Control Subsystem (ADCS) is a three-axis stabilized spacecraft which requires a control attitude of +/- 1.0 degrees and knowledge attitude of +/- 0.1 degree. Several design aspects are considered in development of attitude control systems for a small satellite, such as: spacecraft dynamics, space environment, disturbance torques, orbit type, and spacecraft complexity. The ideal spacecraft's attitude sensor is a rate gyroscope, which provides rate information to the attitude control system. In the case of NPSAT-1, due to budget constraints alternative sensors will be utilized, such as: a three-axis magnetometer, earth sensors, and a Global Positioning System (GPS). A small satellite designed to have a three-axis stabilized, biased momentum system, must have a robust control system, and requires a momentum wheel to provide stiffness to maintain attitude, and magnetic torque rods on each axis. The current design of NPSAT-1 uses all of these sensors to provide rate information for damping and stability to the control system that requires a complicated attitude control design. The purpose of this attitude control design simulation is to investigate and propose a control law utilizing a single pitch momentum wheel and three magnetic torque rods. A further proposal is to utilize a constant speed momentum wheel to avoid momentum damping and over speed, replace the pitch control with magnetic torquers, and develop a Kalman filter estimator to provide all the required angular rates.

THIS PAGE INTENTIONALLY LEFT BLANK

DISCLAIMER

The reader is cautioned that computer programs developed in this simulation may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered completely validated. Any application of these programs without additional verification is at the risk of the user.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF SYMBOLS

The following list includes the definition of major terms used in equations throughout this paper.

μ	Gravitational Constant of the Earth
i	Orbit Inclination
r_p	Radius of Perigee
r_a	Radius of Apogee
e	Orbit Eccentricity
p	Semi-Latus Rectum
Π	Orbit Period
ν	True Anomaly
Ω	Orbital Angular Velocity
ϕ	Roll Error
$\dot{\phi}$	Roll Rate
θ	Pitch Error
$\dot{\theta}$	Pitch Rate
ψ	Yaw Error
$\dot{\psi}$	Yaw Rate
$\vec{T}_{gg}$	Gravity Gradient Disturbance Torque Vector
$\vec{T}_{sp}$	Solar Pressure Disturbance Torque Vector
I	Principal Moment of Inertia Tensor
$\vec{h}$	Reaction Wheel Angular Momentum Vector
k_v	Velocity Feedback Gain
k	Position Feedback Gain
ω_n	Natural Frequency
ζ	Damping Factor
ϕ_{ss}	Steady State Roll Error
θ_{ss}	Steady State Pitch Error
ψ_{ss}	Steady State Yaw Error
A	Plant Matrix
B	Control Matrix
$\vec{u}$	Control Input Vector
F	Control Gain Matrix
$\vec{u}_d$	Disturbance Torque Vector Forcing Function
Φ_k	State Transition Matrix

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

The author would like to thank the following people for their invaluable guidance and assistance provided in the completion of this thesis: Professor Hal Titus, Professor Brij Agrawal, and Professor Barry Leonard, all of the Naval Postgraduate School, Monterey, California.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Design requirements for a small satellite (NPSAT-1) Attitude Determination and Control Subsystem (ADCS) is a three-axis stabilized spacecraft that requires a control attitude of +/- 1.0 degrees and knowledge attitude of +/- 0.1 degree. Several design aspects are considered in development of attitude control systems for a small satellite: spacecraft dynamics, space environment, disturbance torques, orbit type, and spacecraft complexity. The ideal spacecraft's attitude sensor is a rate gyroscope, which provides rate information to the attitude control system. In the case of NPSAT-1, due to budget constraints alternative sensors will be utilized: a three-axis magnetometer, earth sensors, and a Global Positioning System (GPS). A small satellite designed to have a three-axis stabilized, biased momentum system must have a robust control system and requires a momentum wheel to provide stiffness to maintain attitude and magnetic torque rods on each axis. The current design of NPSAT-1 uses all of these sensors to provide rate information for damping and stability to the control system that requires a complicated attitude control design. The purpose of this attitude control design simulation is to investigate and propose a control law utilizing a single pitch momentum wheel and three magnetic torque rods. A further proposal is to utilize a constant speed momentum wheel to avoid momentum damping and over speed, replace the pitch control with magnetic torquers, and develop a Kalman filter estimator to provide all the required angular rates.

Specific to problems targeted by the simulations, external disturbance moments will cause errors in the spacecraft's attitude. These errors will be kept within the required pointing limits if the attitude control system is properly designed. The five major disturbance moments worth consideration are 1) Solar Pressure, 2) Gravity Gradient, 3)

Magnetic Moment, 4) Aerodynamic, and 5) Internal Inertia and Torque. For the purposes of this paper, both magnetic and aerodynamic moments will be discounted. For design simplicity, it will be assumed that the solar pressure moment can be modeled as a constant torque about each body axis.

General design of the NPSAT-1 will be a three-axis stabilized, biased momentum system implementing four sensors and two actuators. Because the spacecraft will be Earth oriented in a polar orbit, attitude control about all three axes is determined to be within one degree (+/-1.0º). The attitude knowledge, however, must be within 0.1 degrees (+/-0.1º) with jitter minimized to less than one milli-radian at 1-kHz. Navigation will be accomplished onboard via a GPS receiver, and external spacecraft guidance is not anticipated.

NPSAT-1 is designed with four information-gathering sensors and two actuators to guide and control its movement. These sensors are: 3-axis magnetometer, 3-axis star sensor, earth sensors, and a GPS receiver. These four sensors are defined to gather information to feed the information stream supporting the Kalman filter and the ADCS.

In addition to the four sensors, NPSAT-1 is designed with two actuators to guide and control its movement. These are the Pitch Momentum Wheel and the Magnetic Torquers. A single momentum wheel will be placed along the pitch axis of the NPSAT-1. Angular momentum generated by the spinning wheel will provide gyroscopic stability to the spacecraft about the roll and yaw axes. Pitch will be controlled by spinning the wheel up or down. Roll and yaw as well as momentum dumping is to be controlled by the torque rods via control laws. A set of three torquers, one on each axis, will be

employed to initially orient the NPSAT-1. They also are to be used to spin up the pitch momentum wheel for momentum dumping and to control roll and yaw via control laws.

The results obtained in this thesis are quite extraordinary. The controller uses a magnetic torque actuator to create the required torques. The linear principle of superposition allowed removal of wheel speed changing, creating a constant speed wheel. The system was well behaved. The orbit inclination is also a concern. This approach will probably have problems with equatorial or polar orbits.

This thesis shows that a properly designed optimal rate estimator Kalman filter is effective and able to estimate body rate from a single star sensor. In addition, the initial results prove that a single sensor coupled with a proper rate estimator design can be used as a backup, or even primary, attitude determination process.

For NPSAT-1, a single star sensor estimator will be an addition to the control system. This approach is necessary, especially during the initial launch, where after initial launch the spacecraft will be tumbling at some pitch, roll, and yaw rates. The magnetometer and the magnetic torquers control, but would require, an additional vector which the star sensor can provide.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The micro-satellite space industry has grown at an alarming rate over the last decade and will continue to develop into an advanced and sophisticated space marketplace. The majority of small satellites in orbit or under development were designed and manufactured by universities in collaboration with Aerospace Corporation.

To maximize revenue, today's satellites are designed to have extended lifetimes and precise attitude control systems. The majority of satellites in orbit is exceeding their life cycles and operating beyond their designed lifetime. However, some of the satellite components are developing problems directly related to the degradation of their critical components, such as the gyroscopic rate sensors. These rate sensors are a critical element that is vital to overall functioning and provides an attitude determination and control system (ADCS) to the spacecraft. To counter this problem, rate estimator development is needed from different sensor sources that fuse multiple data streams together for use as an attitude determination and control system.

Many factors can shorten the lifetime of a satellite, such as: space environment, disturbance torques, orbit type, and spacecraft complexity. A spacecraft becomes complex when deployment mechanisms and moving parts are incorporated into its design. A rate gyroscope, for example, is a constantly spinning piece of equipment, which provides rate information to the attitude determination and control system. By way of example, if a satellite is designed to have an operational life of 15 years, the reliability of its rate gyroscope decreases over the operational lifetime, more so than a non-moving or solid-state piece of hardware. Rate information provides robustness,

damping, and stability to a control system. If this stability is lost, a secular disturbance torque could conceivably destabilize the satellite [1].

In the project that this paper is based on, students at the Naval Postgraduate School designed a micro-satellite functioning as a space-borne platform for multi-spectral imaging. NPS Space Systems Engineering students designed the satellite for the Naval Postgraduate Satellite program (NPSAT-1) as a research, design, fabrication, testing, integration, launch, and on-orbit implementation of a Low Earth Orbiting (LEO) satellite. The satellite's mission was conceptualized in support of a science payload dedicated to multi-spectral imaging of the earth's Aurora.

This paper will discuss in some detail the design of two of the main components of the NPSAT-1, namely, the Attitude Dynamics Control System and Kalman Filter Rate Estimator. The discussion of these components will include their conceptualization, theoretical origins, tests and simulations, a presentation of the results, conclusions drawn from the results and how they apply to the final design of NPSAT-1, and, suggestions for future research.

A. SATELLITE FAILURE ISSUES AND IN-TIME SOLUTIONS

Many satellites have been developed since Explorer I was launched in 1959, which perform a multitude of missions. Mission possibilities include: communications, mapping, and weather observation. Some of these satellites are spin-stabilized, dual spin-stabilized, and three-axis stabilized. Due to the growing need for power, most of today's spacecraft are three-axis stabilized and will be the type of spacecraft considered in this study. Typically, communication satellites are in a near geo-stationary orbit for middle-to-low latitude coverage, and in a Molniya orbit for high latitude coverage.

As we move into the 21st century, we become increasingly dependent on satellites. If a spacecraft fails before its design life runs out, the resulting loss in revenue can be staggering. Although many factors can contribute to spacecraft failure, the most common limiting factors are fuel, batteries, and solar cell degradation. This, of course, is highly dependent upon the altitude and mission of the satellite. For long duration missions, it is not uncommon for a rate gyroscope to fail. This can be a critical failure if the satellite's attitude control laws require inputs from this device.

Operational information and data received from a satellite are generally ignored until a critical outage occurs. As an example, the unexpected and premature failure of the Galaxy IV spacecraft temporarily left millions of people without pager service. This failure translates into a multi-million dollar loss of financial revenue for businesses highly dependent on this technology. That is why when a satellite fails and its life expectancy is threatened, every effort will be made to save it.

Recently, the SOHO spacecraft lost its orientation after suffering from multiple gyroscope failures. With that, engineers were forced to create a software package that would override the failed hardware. It took nearly six months to write and test the code before the control system of the billion-dollar satellite was restored as a successful uplink. SOHO is now able to autonomously maintain proper attitude relative to the sun using its star tracker as the primary control sensor. SOHO is testimony that after a certain hardware failure there will be a great need to develop software upgrades to current systems on orbit in order to extend its life cycle.

A part of this research is to devise a reliable method of obtaining angular rates from a star tracker in the event of a gyroscope failure. Since star sensors only measure

errors in angle, rates will have to be derived based on these measurements. At first glance, it might seem reasonable to simply take the derivative of the angles to get the required rate information, but doing so would only amplify the noise effects of the sensor. Pseudo-rate modulators can be used to derive rates, but the accuracy of these modulators is strictly a function of sensor noise. A Kalman filter however, can determine angular rates as well as reduce noise, which is inherent in any sensor.

Spacecraft attitude determination algorithms traditionally relied only on rate gyros. These gyros are highly reliable but deteriorate over time and degrade system performance dramatically as evident in the complete replacement of the gyros onboard the Hubble Space Telescope. Recent advances in star trackers have allowed several research projects to develop Kalman filter algorithm simulations. Such research utilizes the single most accurate instrument on spacecraft thus far, the star sensor.

B. OVERVIEW OF NPSAT-1 ATTITUDE DYNAMICS CONTROL SYSTEM DESIGN

The design of the Attitude Control System for NPSAT-1 was developed by students during the preliminary design phase of AA4871 Spacecraft Design II Engineering, with the assistance of Professor Barry Leonard. During this phase, certain design criteria were specified based on the mission requirements of the spacecraft. The proposed control law, modeling, and simulation using MATLAB were developed during AA3818 Spacecraft Attitude, Dynamics, and Controls, as a design project with the assistance of Professor Barry Leonard. The proposed MATLAB's SIMULINK block diagram of the attitude control system is shown below in Figure 1.

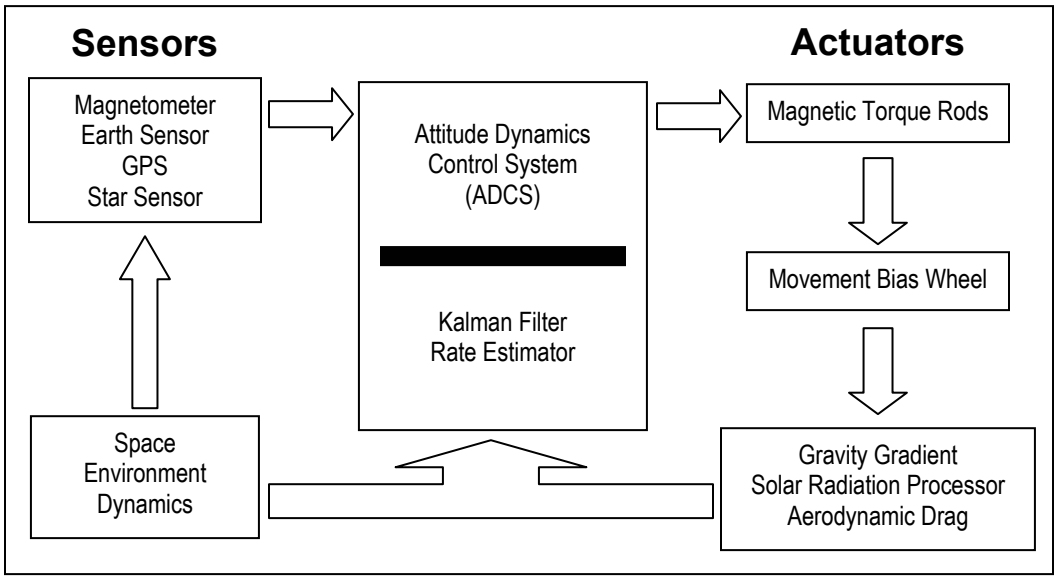


Figure 1: Attitude Control System (Systems Design Diagram)

The goal of the project described in this paper required the proposal of a control law that can substitute magnetic torquers that use the Earth's magnetic field in place of reaction jets.

The following steps were required:

- Add the magnetic field in orbital coordinates to the simulation.
- Translate the field components into body coordinates.
- Assume "ideal" magnetic torquers m_x, m_y, and m_z followed by a saturation characteristic.
- Form the cross product of $\mathbf{m} = (m_x, m_y, m_z)^T$ with the **B** body coordinates to generate the control torques $\mathbf{M}_c = (M_{xc}, M_{yc}, M_{zc})^T$.
- Disconnect the reaction jet control torques from the spacecraft and replace it with the magnetic control torques.
- Derive the magnetic torquers with the following control laws:
  - $m_x = 0$
  - $m_y = 0$
  - $m_z = 0$

- Remove the disturbance torques $\mathbf{M} = (M_{dx}, M_{dy}, M_{dz})^T$ and demonstrate acquisition from $\theta = 5°$, $\phi = 5°$, $\psi = 5°$. If possible, acquisitions from large angles and perturbation torques should be performed.

C. DESIGN OF KALMAN FILTER RATE ESTIMATOR

In 1960, R. E. Kalman published his famous paper describing a recursive solution to the discrete-data linear filtering problem. Since that time, due in large part to advances in digital computing, the Kalman filter has been the subject of extensive research and application, particularly in the area of autonomous or assisted navigation.

In this paper, the Kalman filter will be applied to a rate estimator that is integral to controlling a small satellite. The Kalman filter is an important component in satellite navigation because it has the potential to determine angular rates as well as reduce noise, which is inherent in any sensor. A major part of this project, and as demonstrated in this paper, is the simulation of the Kalman filter and its interaction with the navigational components of the small satellite [2].

D. LITERATURE REFERENCE NOTE

The primary literature source for this paper is the "NPSAT-1 Preliminary Design Report" dated September 15, 1999 [1] and noted in the list of references at the end of this paper. Because of the pervasive use of this document throughout the project, the author does not explicitly cite each reference to this report.

II. SPACE ENVIRONMENT

The space environment consists of all things related to being in space, such as radiation, the effects of the sun, and degradation of the satellite system itself. A combination of the satellite's mission and the space environment itself aid in determining the satellite's orbit type. Key factors in selecting a satellite's orbit type is its altitude selection based on the satellite's radiation environment and mission purpose. More precisely, the satellite's radiation environment undergoes a substantial change at about 1000 kilometers. Below this altitude, the atmosphere will quickly clear out charges particles so the radiation density is relatively low. Above this altitude are the Van Allen belts of trapped radiation that can significantly reduce the lifetime of components.

Mission orbits can therefore be separated into two types: low earth orbits (LEO) below 1000 km. and geosynchronous orbits (GEO) that are above 1000 km. Because the NPSAT-1 is designed to study the Earth's aurora, a low earth orbit appears to be the most practical choice.

Orbit selection is a trading process balancing the satellite's mission assignment against such factors as payload size and weight, launch cost, and design lifetime. Once we have selected the LEO orbit category as being the most reasonable, the orbit itself can be refined further. There are two possibilities in orbit style that may be attractive: a standard LEO that is almost circular, and a Molniya orbit that is semi-synchronous and eccentric. The advantage of an eccentric orbit is that at apogee the satellite's velocity is lower and offers more time at apogee. Since we are studying the Earth's aurora, this might be an attractive solution, as the satellite would have more time to record data if

positioned at apogee above the geographical area of the aurora. These are the two orbit solutions covered in this research. For purposes of clarity, both orbits are earth referenced.

A. DISTURBANCE TORQUES

External disturbance moments will cause errors in the spacecraft's attitude. These errors will be kept within the required pointing limits if the attitude control system is properly designed. The five major disturbance moments worth consideration are 1) Solar Pressure, 2) Gravity Gradient, 3) Magnetic Moment, 4) Aerodynamic, and 5) Internal Inertia and Torque. For the purposes of this paper, both magnetic and aerodynamic moments will be discounted. For design simplicity, it will be assumed that the solar pressure moment can be modeled as a constant torque about each body axis. These moments were found to be [2],

$$\begin{aligned} T_{ggx} &= 3\Omega^2 \phi\left(I_z - I_y\right), \\ T_{ggy} &= 3\Omega^2 \theta\left(I_z - I_x\right), \\ T_{ggz} &= 0 . \end{aligned} \tag{1}$$

Note that the symbols in Equation (1) as well as all other symbols used in this paper are defined in the section LIST OF SYMBOLS detailed in the front-matter of this paper.

1. Modeling Disturbance Torques

The following disturbance torques was estimated using SMAD. The total worst-case disturbance torque was used to size the actuator components. Additionally, a MATLAB and SIMULINK model of the ADCS sub-system that was generated took into account all of the disturbances discussed in the following subsections.

a) Magnetic

The Magnetic Disturbance Torque is cyclic throughout orbit. The magnetic torque is altitude dependent with higher torque at lower altitudes. The torque can be estimated using the following equation,

$$T_m = DB = D\left(2M/R^3\right), \tag{2}$$

where,

D = Residual Dipole of the Spacecraft = 2 A-m^2
B = Earth's Magnetic Field ~ 2M/R^3 (for a near-polar orbit)
M = Earth's Magnetic Moment = 9.00 x 10^{15} Tesla-m^3
R = Radius of Orbit = 6878 x 10^3 m (Altitude = 500 km)

The worst-case Magnetic Field Torque was calculated to be 1.11 x 10^{-4} N-m.

b) Aerodynamic

The Aerodynamic Torque can be estimated using the following equation. This torque is altitude dependent and is at its worst for lower altitudes,

$$T_a = 0.5\rho C_d A V^2 (C_{pa} - C_m), \tag{3}$$

where,

ρ = Atmospheric Density = 2.8 x 10^{-12} kg/m^3
C_d = Coefficient of Drag = 2.5
A = Surface Area of Spacecraft = 1.0 m^2
V = Spacecraft Velocity = 7613 m/s
($C_{pa} - C_m$) = Center of Aerodynamic Pressure/Mass offset = 0.2 m

The worst-case Aerodynamic Torque was calculated to be 1.52 x 10^{-5} N-m.

c) Solar Radiation

Solar Radiation Torque is a cyclic torque that varies throughout the orbit. This torque, however, was not dependent on altitude. It was estimated using the following equation,

$$Tsp = \left(\frac{F_s}{c} \right) A_s (1+q) \cos(i) (C_{ps} - C_m), \tag{4}$$

where,

F_s = Solar Constant = 1399 W/m^2
c = Speed of Light = 3 x 10^8 m/s
A_s = Spacecraft Surface Area = 1.0 m^2
q = Reflectance Factor = 0.7
i = Angle of Incidence of Sun = 0 (worst case)
($C_{ps} - C_m$) = Center of Solar Pressure/Mass offset = 0.2 m

Using the worst-case values above, we obtain the Solar Radiation Torque to be a maximum of 1.59 x 10^{-6} N-m.

d) Gravity Gradient

In a Molniya orbit, gravity gradient moments will be greatest at perigee. The Gravity Gradient Torque is given by,

$$\vec{T}_{gg} = \int \vec{r} x \vec{a}_g dm \, . \tag{5}$$

The gravitational acceleration is,

$$\vec{a}_g = -GM_{\oplus} \frac{\vec{R} + \vec{r}}{\left| \vec{R} + \vec{r} \right|^3} \, . \tag{6}$$

$\vec{R}$ is the distance to the center of mass of the satellite measured from the center of the Earth and it is given by,

$$\vec{R}_o = R \hat{o}_3 \, . \tag{7}$$

Equation (7) expressed in body coordinates is,

$$\vec{R}_b = {}^b C^o \vec{R}_o \, . \tag{8}$$

For now, $\vec{R}_b$ will be written as,

$$\vec{R}_b = X \hat{b}_1 + Y \hat{b}_2 + Z \hat{b}_3 \, . \tag{9}$$

Taking the cross product, we obtain

$$\vec{r}x\vec{R} = (yZ - zY)\hat{b}_1 + (zX - xZ)\hat{b}_2 + (xY - yX)\hat{b}_3 . \tag{10}$$

From the binomial theorem, the following expression is obtained

$$\left|\vec{R} + \vec{r}\right|^{-3} = \frac{1}{R^3} - 3\frac{xX + yY + zZ}{R^5} . \tag{11}$$

It can be shown that the orbital angular velocity is just

$$\Omega = \sqrt{\frac{GM_{\oplus}}{R^3}} . \tag{12}$$

Equations (9), (10), and (11) can be substituted into Equation (12) to get the following expression

$$\begin{aligned} T_{ggx} &= -3\Omega^2[\phi(I_y - I_z) - \theta I_{xy} - I_{yz}] , \\ T_{ggy} &= -3\Omega^2[\theta(I_x - I_z) + \phi I_{xy} + I_{xz}] , \\ T_{ggz} &= 3\Omega^2(\phi I_{xz} + \theta I_{yz}) . \end{aligned} \tag{13}$$

These three equations were derived using small angle approximations and ignoring second order terms.

B. ATTITUDE DYNAMICS CONTROL SYSTEM (ADCS) GUIDANCE AND CONTROL CONCEPT

The NPS Aurora Satellite (NPSAT-1) requires a scheme to sense disturbances and a method of reacting to disturbance. This section discusses the general design concept for the ADCS.

1. General

The NPSAT-1 will be a three-axis stabilized, biased momentum system implementing four sensors and two actuators. Because the spacecraft will be Earth oriented in a polar orbit, attitude control about all three axes is determined to be within one degree (+/-1.0º). The attitude knowledge, however, must be within 0.1 degrees (+/-0.1º), with jitter minimized to less than one milli-radian at 1-kHz. Navigation will be

accomplished onboard via a GPS receiver, and external spacecraft guidance is not anticipated.

2. Total Worst-Case Disturbance

The worst-case external disturbance torque is calculated as the sum of the above external disturbance torques. While this method may result in an over design of the ADCS, it accounts for the low probability event of all worst-case external torques occurring simultaneously and in the same direction. Therefore, the total worst-case disturbance torque for NPSAT-1 is,

$$T_d = T_m + T_a + T_{sp} = 1.04 \text{ x } 10^{-4} \text{ N-m.} \quad (14)$$

C. COMPONENTS

NPSAT-1 is designed with four, information-gathering sensors and two actuators to guide and control its movement. This section discusses each of the components required by the design concept for information gathering and control.

1. Sensors

There are four sensors defined to gather information for the NPSAT-1. These are the Magnetometer, Star Sensor, Earth Sensors, and a GPS Receiver.

a) 3-Axis Magnetometer

A three-axis magnetometer will be used to detect Earth's magnetic field to determine the NPSAT-1's attitude to approximately 5-10 degrees accuracy. Output magnetometer sensor will be used as the primary method of attitude determination after separation from the launch vehicle and initially orient the NPSAT-1. Once initial orientation is completed, the Earth sensors will take over for final attitude acquisition. The magnetometer will also contain inputs to the magnetic torquers, to be discussed later.

b) 3-Axis Star Sensor

A star camera will be used to automatically determine the NPSAT-1's attitude in three axes. This will be used as a backup method of initial acquisition. The star sensor is limited to operate only when pointed at space without interference from the Sun, Earth, Moon, or other objects. Accuracy of the star sensor is well above the requirement of 0.1 degrees and provides yaw information not available from the Earth sensors. The star sensor could provide attitude knowledge a majority of the time, in and out of eclipse, while a sun sensor would not operate.

c) Earth Sensors

Two sets of static infrared horizon sensors will provide the primary control input to the NPSAT-1's actuators. The first set will be 'coarse' sensors with a field of view of 20°. This set will be used the for Earth horizon acquisition following initial orientation guidance from the magnetometers. The second set of horizon sensors are 'fine' sensors and will be used for the primary control of the NPSAT-1's attitude. Both sets of horizon sensors provide roll and pitch knowledge. The coarse sensors are set to within 1.15°, while the fine sensors are set to within 0.15° accuracy.

d) GPS Receiver

A GPS receiver is to be used for obtaining NPSAT-1 location information. The receiver would aid the user in locating the satellite, and hence, the observables from the payloads. The altitude of the NPSAT-1 is low enough that GPS satellites will be accessible to the GPS receiver. Signals from this system will give ground controllers exact position and velocity of the satellite, thus providing accurate orbital information. If GPS technology allows, it might be possible to gain backup attitude information from differential GPS through judicious placement of 4 to 5 GPS antennas. For example, an

antenna could be placed on the zenith face and on each tip of the unfolding solar array. Theoretically, this would provide the necessary spacing for spacecraft attitude to be calculated.

2. Actuators

In addition to the four sensors, NPSAT-1 is designed with two actuators to guide and control its movement. These are the Pitch Momentum Wheel and the Magnetic Torquers. A discussion of these actuators is the topic of this section.

a) Pitch Momentum Wheel

A single momentum wheel will be placed along the pitch axis of the NPSAT-1. Angular momentum generated by the spinning wheel will provide gyroscopic stability to the spacecraft about the roll and yaw axes. Pitch will be controlled by spinning the wheel up or down. Roll and yaw as well as momentum dumping is to be controlled by the torque rods via control laws.

b) Magnetic Torquers (Torque Rods)

A set of three torquers, one on each axis, will be employed to initially orient the NPSAT-1. They also are to be used to spin up the pitch momentum wheel, for momentum dumping, and to control roll and yaw via control laws. The torque rods are to be double wound for redundancy.

D. ATTITUDE CONTROL SYSTEM DESIGN SPECIFICATIONS

The development of Spacecraft Attitude, Dynamics, and Controls design is to build a suitable attitude control law. The attitude control system will be designed according to the following parameters.

1. Satellite Specifications

- LEO orbit

- Two earth sensors aligned with the body axes
- One star sensor
- Three-axis magnetometer
- Global Positioning System (GPS)
- Roll inertia, I_{xx}=24.67 kg-m^2
- Pitch inertia, I_{yy}=22.63 kg-m^2
- Yaw inertia, I_{zz}=11 kg-m^2
- 550 km altitude, circular orbit
- Kalman filter rate estimator (future development)

2. Other Specifications

- Nadir pointing to within +/- 1°
- 4 arc-second noise level for each sensor
- Three-axis stabilized
- 1-year design life

3. Assumptions

- Small and large angle (up to 10 r/s) approximations
- Orbital angular velocity and acceleration known for each sensor measurement
- Constant solar pressure moments
- No slewing requirement
- Satellite is modeled as a rigid body

4. Attitude Control System Design Considerations

In order to achieve 1.0° pointing accuracy, a constant speed momentum wheel, and three magnetic torquers whose momentum vectors coincide with the body axes will

be employed. As disturbance moments cause errors in attitude, off-axis components of reaction wheel angular momentum will cause internal torques that must be accounted for.

III. MODELING AND SIMULATION

Many types of kinematical transformation methods are in use in various types of research, but the most popular are: direction cosine matrices (DCM), Euler angles, and quaternion. Quaternion is popular since it involves only a single rotation about an Eigen-axis. On the other hand, making small angle approximations and setting second order terms to zero, DCM's are easily employed and are used in this analysis. Transformations from one frame to another are performed to facilitate calculations. For example, the latitude and longitude of stars in the star catalog have all been programmed within a celestial frame, but measurements will be made in the body frame. Therefore, proper attitude determination relies on a simple transformation [3].

In the field of attitude control, it is often required to express an inertial quantity as a body frame quantity. For example, the inertial angular velocity derived from the Euler moment equations must be expressed in body coordinates, and then integrated to get the Euler angles. The body frame, orbital frame, and inertial frame are the three reference frames are used in the derivation of equations of motion:. The origin of these three frames will all be located at the spacecraft's center of mass. In the right-hand set, the orbital reference frame, the Z-axis points at the center of the Earth, the X-axis points in the satellite's direction of motion, and the Y-axis is normal to the orbital plane, completing. In the left-hand set, the body reference frame is attached to the spacecraft; therefore, the Euler angles represent the deviation of the body reference frame from the orbital reference frame. On-board sensors measure these Euler angles. The inertial frame remains fixed in Earth space such that the inertial Y-axis coincides with the orbital

Y-axis. The celestial frame is an additional reference frame alluded to earlier. The Z-axis of this frame points north and the X-axis points in the direction of the vernal equinox. Although the X-axis precesses (albeit very slowly), the assumption is that it is fixed in space.

A. STAR SENSOR CHARACTERISTICS

The star sensor model used in this simulation was designed in accordance with the specifications outlined in Chapter II. Table 1 summarizes the characteristics of the star sensor and, for completeness; additional assumptions have been made that will be consistent with current technology.

Technology	Charged Coupled Device (CCD)
FOV	10° x 10°
Accuracy	~10 arcsec
# Stars in Catalog	4000
Sampling Rate	0.1 Hz (current technology is faster)
Noise	4 arcsec (magnitude=6)
Solar Exclusion Angle	30° w/sun shade

Table 1: Star Tracker Characteristics.

The noise level shown in Table 1 is inherent to the star tracker itself and it is treated as a zero-mean Gaussian white sequence.

B. GPS SYSTEMS

Three-axis attitude determination requires two separate line-of-sights (LOS) with angular separation near 90° for increased accuracy. In this simulation, the optical axis of each star sensor will be aligned with the body axes, and at each discrete time step a star sensor will be selected at random for attitude determination. Although this sequence of events permits only one LOS per time step, attitude is readily determined over consecutive time steps. Since only one star will be in the sensor FOV at any particular time, measurements can only be made about two axes.

C. REFERENCE FRAMES

The simulation discussed in this section makes use of three previously discussed frames of reference; the body frame, the inertial frame, and the orbital (LVLH) frame. A fourth frame of reference, the earth frame, is introduced to complete the simulation. Orientation of the frames is as follows; the body frame is fixed to the spacecraft and aligned with the satellite's principal moments of inertia, the inertial frame is celestially fixed, while the angular velocities of the orbit and earth frames with respect to the inertial frame, respectively, are given by [5]

$$\begin{aligned} {}^{i}\vec{\omega}^{o} &= -[{}^{b}C^{o}]\Omega_{o}\ \hat{o}_2, \\ {}^{i}\vec{\omega}^{e} &= [{}^{b}C^{e}]\Omega_{e}\ \hat{e}_3. \end{aligned} \tag{15}$$

The angular velocity of the body frame with respect to the orbital frame is given by

$${}^{o}\vec{\omega}^{b} = \dot{\phi}\hat{b}_1 + \left[{}^{b}C^{n}\right]\dot{\theta}\hat{n}_2 + \left[{}^{b}C^{o}\right]\dot{\psi}\hat{o}_3, \tag{16}$$

where n is just an intermediate frame. Since on-board sensors make measurements with respect to the body-frame, all of the above rotation rates will be transformed to the body-frame. As can be seen from Equations (15) and (16), the C matrices (DCM), perform this transformation. For example, the DCM that transforms the orbital frame to the body frame is given by the following 3-2-1 rotations

$${}^{b}C^{o} = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ -c\phi s\psi + s\phi s\theta c\psi & c\phi c\psi + s\phi s\theta s\psi & s\phi c\theta \\ s\phi s\psi + c\phi s\theta c\psi & -s\phi c\psi + c\phi s\theta s\psi & c\theta c\phi \end{bmatrix}. \tag{17}$$

Expanding Equation (16) and getting rid of 2nd order terms,

$${}^{o}\vec{\omega}^{b} = \left(\dot{\phi} - \dot{\psi}s\theta\right)\hat{b}_1 + \left(\dot{\theta}c\phi + \dot{\psi}s\phi c\theta\right)\hat{b}_2 + \left(\dot{\psi}c\theta c\psi - \dot{\theta}s\phi\right)\hat{b}_3. \tag{18}$$

From Equation (18), the Euler rates can now be determined,

$$\begin{aligned} \dot{\phi} &= \omega_1 + \omega_2 \tan\theta \sin\phi + \omega_3 \tan\theta \cos\phi \,, \\ \dot{\theta} &= \omega_2 \cos\phi - \omega_3 \sin\phi \,, \\ \dot{\psi} &= \frac{\omega_2 \sin\phi + \omega_3 \cos\phi}{\cos\theta} \,. \end{aligned} \tag{19}$$

These rates can be integrated to give the Euler angles. These angles represent the deviation of the body frame with respect to the orbit frame. The fine horizon sensor will detect these errors and feed them to the momentum wheel that, in turn, will act as an actuator to correct this error. The magnetometer and the star sensor will act in a similar fashion; however, the star sensor will not be part of the control loop. Instead, its attitude information will be sent down to the Alaska ground station for attitude knowledge. The operational purpose of this attitude control system is to keep the body frame aligned with the orbital frame. The following table lists the characteristics of this 3-axis stabilized micro-satellite.

Altitude	550 km
Period	5677.0 s
Ω_o	0.0011068 rad/s
I_{xx}	22.222 kg-m^2
I_{yy}	21.387 kg-m^2
I_{zz}	17.056 kg-m^2
Pointing Knowledge	.1° all axes
Pointing Accuracy	1° all axes

Table 2: Satellite Properties

D. SATELLITE DYNAMICS

Since it is assumed that the spacecraft rotates about its principal moments of inertia, the spacecraft's angular momentum is given by [6]

$$\vec{H} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} {}^{i}\vec{\omega}^{b} \,. \tag{20}$$

Notice that to use Equation (20), the angular velocity must be inertial and it must be expressed in body coordinates; using small angle approximations, it can be determined by adding Equation (15) and (18)

$$ {}^{i}\vec{\omega}^{b} = \left(\dot{\phi} - \Omega_o \psi\right)\hat{b}_1 + \left(\dot{\theta} - \Omega_o\right)\hat{b}_2 + \left(\dot{\psi} + \Omega_o \phi\right)\vec{b}_3 . \tag{21} $$

It is also important to note that *H*, in Equation (20), is the total angular momentum of the satellite in expressed in body coordinates, including the momentum wheel. It can be broken up as follows

$$ \vec{H} = \vec{H}^{b} + \vec{h}_w . \tag{22} $$

Each angular momentum component rotates about its own center of mass, and the momentum wheel only rotates about the negative pitch axis, so that $\vec{h}_w = -h\hat{b}_2$. The inertial rate of change of angular momentum is just equal to the external moment

$$ \vec{M} = \frac{{}^{i}d}{dt}\vec{H} = \frac{{}^{b}d}{dt}\vec{H} + {}^{i}\vec{\omega}^{b} \times \vec{H} . \tag{23} $$

The external moment, M, in Equation (23) represents the sum of all of the external moments, including gravity gradient, aerodynamic, solar radiation pressure, magnetic control, and pitch wheel control. The gravity gradient moment is given as Equation (16)

$$ \vec{M}_{gg} = 3\Omega_o^2 \phi (I_{zz} - I_{yy})\hat{b}_1 + 3\Omega_o^2 \theta (I_{zz} - I_{xx})\hat{b}_2 . \tag{24} $$

The other moments will be derived later. Substituting Equations (21), (22), and (24) into Equation (23), the following results are obtained

$$ \begin{aligned} M_{dx} &= I_{xx}\ddot{\phi} + \left[-4\Omega_o^2\left(I_{zz} - I_{yy}\right) + \Omega_o h\right]\phi + \left[-\Omega_o\left(I_{xx} - I_{yy} + I_{zz}\right) + h\right]\dot{\psi} , \\ M_{dy} &= I_{yy}\ddot{\theta} - 3\Omega_o^2\left(I_{zz} - I_{xx}\right)\theta + \dot{h} , \\ M_{dz} &= I_{zz}\ddot{\psi} + \left[\Omega_o^2\left(I_{yy} - I_{xx}\right) + \Omega_o h\right]\psi + \left[\Omega_o\left(I_{xx} - I_{yy} + I_{zz}\right) - h\right]\dot{\phi} . \end{aligned} \tag{25} $$

As can be seen from Equation (20), the pitch moment equation is independent of roll and yaw. These moment equations are in agreement with Equation (15).

E. PITCH CONTROL

The solution to Equation (25) is oscillatory, if $\dot{h}$=0, i.e. no wheel control, the following occurs

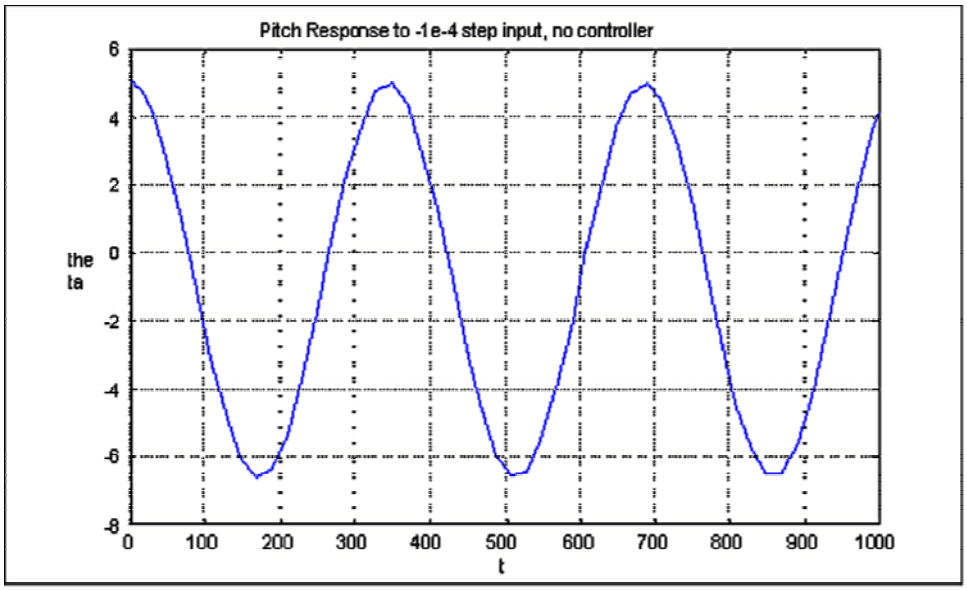


Figure 2: Pitch response with no damping.

The response in Figure 2 can be dampened out by controlling the momentum wheel, $\dot{h}$, according to the following control law

$$\dot{h} = k_\theta \tau_\theta \dot{\theta} + k_\theta \theta \,. \tag{26}$$

Substituting Equation (26) into Equation (25) and taking the Laplace transform, the following transfer function is derived

$$\frac{\Theta(s)}{M_{dy}(s)} = \frac{1/I_{yy}}{s^2 + \frac{k_\theta \tau_\theta}{I_{yy}} s + \frac{k_\theta + 3\Omega_o^2 (I_{xx} - I_{zz})}{I_{yy}}} \,. \tag{27}$$

The plot in Figure 2 assumes an initial condition of θ = 5°, but Equation (27) is based on the initial conditions being zero. It is of no consequence, however, since the initial conditions will not affect the characteristic equation. The steady state error can be

determined for a step input disturbance torque using the final value theorem Equation (19) and it can be shown that

$$\theta_{ss} = \frac{M_{dy}}{I_{yy}\omega_{\theta}^2}. \tag{28}$$

The allowable steady state error given by the requirements in Table 2 is ±1°, but .8° will be chosen as the design margin. In addition, the constants in Equation (41) will be determined based on a critically damped system. The following table lists the pitch control properties.

steady state error, θ_{ss}	±.8°
worse case pitch disturbance moment, M_{dy}	-.0001 N-m
damping ratio,ξ	1
natural frequency, ω_{θ}	0.0182995 rad/s
pitch time constant, τ_{θ}	109.0031 s
pitch auto-pilot gain, k_{θ}	0.00718096 N-m/rad

Table 3: Pitch Controller Properties

Using the controller gains given in Table 3, the following response is obtained

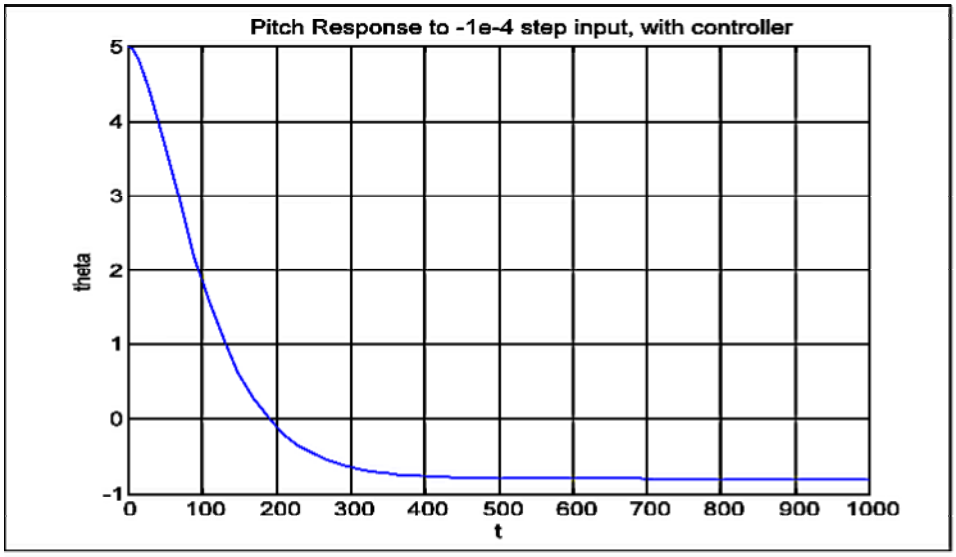


Figure 3: Pitch response with damping.

F. ROLL-YAW RESPONSE

The horizon sensor is capable of detecting both pitch and roll errors, so both of these angles will be available for state feedback. The momentum wheel gives the spacecraft gyroscopic stiffness along the pitch axis, so that when a roll or a yaw error occurs, the result is mutation about the pitch axis, similar to a spinning top. Assuming that $h >> \max(I\Omega_o)$, it can be shown that Equations (25) can be simplified to

$$\begin{aligned} M_{dx} &= I_{xx}\ddot{\phi} + h\Omega_o\phi + h\dot{\psi}\,, \\ M_{dz} &= I_{zz}\ddot{\psi} + h\Omega_o\psi - h\dot{\phi}\,. \end{aligned} \tag{29}$$

Taking the Laplace transform of Equations (29) the result is four transfer functions; however, the design transfer functions are the following

$$\begin{aligned} \frac{\Psi(s)}{M_{dz}(s)} &= \frac{I_{ss}s^2 + \Omega_o h}{\left(I_{xx}s^2 + \Omega_o h\right)\left(I_{zz}s^2 + \Omega_o h\right) + h^2 s^s}\,, \\ \frac{\Phi(s)}{M_{dz}(s)} &= \frac{I_{xx}s^2 + \Omega_o h}{\left(I_{xx}s^2 + \Omega_o h\right)\left(I_{zz}s^2 + \Omega_o h\right) + h^2 s^s}\,. \end{aligned} \tag{30}$$

The yaw and roll response of these two transfer functions is shown in Figure 4.

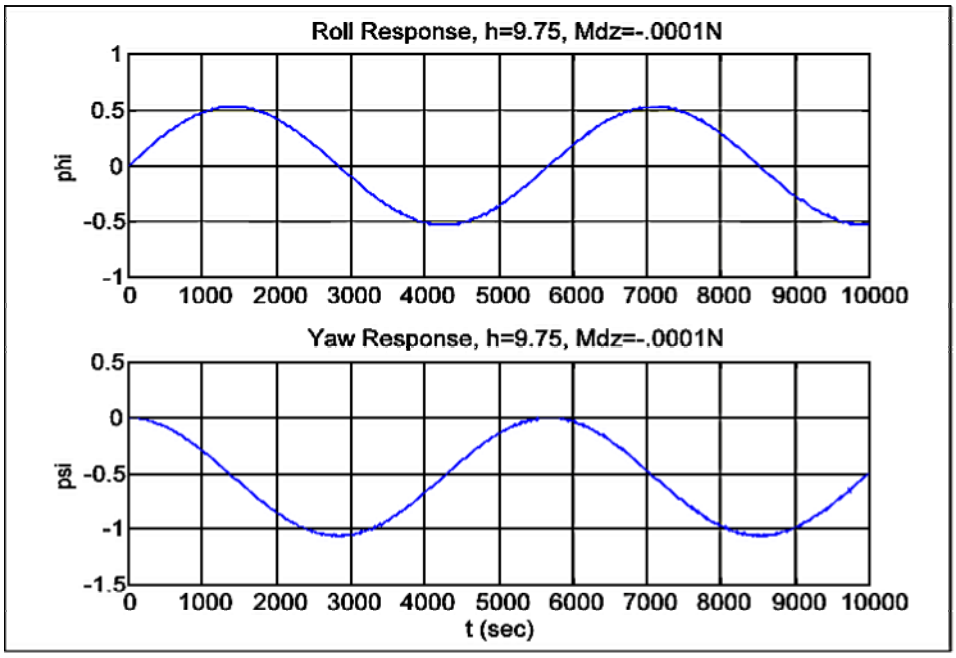


Figure 4: Undamped roll and yaw responses to step input.

The roll response is within the pointing requirements, but the yaw response exceeds 1° after 40 minutes. This is assuming that the initial conditions for roll and yaw are both zero. If the angular momentum of the wheel is increased, both responses would be within limits; for example, if h=15Nms, operationally, no roll-yaw damping would be required. For acquisition, however, roll and yaw damping will be required since there will be an initial error upon launch vehicle separation. It is interesting to note that the period of the responses is simply the period of the orbit; there is also a short period response, which is related to momentum wheel precession. Evidence of roll-yaw coupling can also be seen since the two plots are 90° out of phase.

G. ATTITUDE CONTROL DETERMINATION

Many types of control laws are available, which can conceivably satisfy this satellite's pointing requirements. Some common control laws are: 1) proportional, 2) proportional plus derivative, 3) proportional plus integral plus derivative, and 4) optimal.

Each of these controllers has its own unique characteristics; however, as long as the controllers maintain proper spacecraft attitude, controllers that are more exotic will not be required. In fact, it will be shown that the gains of a simple PD controller can be adjusted to minimize overshoot and settling time,

$$\begin{aligned} \dot{h}_x &= k_{vx}\dot{\phi} + k_x\phi\,, \\ \dot{h}_y &= k_{vy}\dot{\theta} + k_y\theta\,, \\ \dot{h}_z &= k_{vz}\dot{\psi} + k_z\psi\,. \end{aligned} \tag{31}$$

These control laws are expressed as the rate of change of reaction wheel angular momentum, or reaction wheel torque, and they are part of the feedback loop. As can be seen, these internal torque equations are a function of the measured Euler angles and rates. If the resulting set of equations is completely de-coupled, and the Laplace transform is taken, the following result is obtained,

$$\begin{aligned} \frac{\Phi(s)}{T_x(s)} &= \frac{\dfrac{1}{I_x}}{s^2 + \dfrac{k_{vx}}{I_x}s + \dfrac{4\Omega^2(I_y - I_x) - \Omega h_y + k_x}{I_x}}\,, \\ \frac{\Theta(s)}{T_y(s)} &= \frac{\dfrac{1}{I_y}}{s^2 + \dfrac{k_{vy}}{I_y}s + \dfrac{3\Omega^2(I_x - I_z) + k_y}{I_y}}\,, \\ \frac{\Psi(s)}{T_z(s)} &= \frac{\dfrac{1}{I_z}}{s^2 + \dfrac{k_{vz}}{I_z}s + \dfrac{\Omega^2(-I_x + I_y) - \Omega h_y + k_z}{I_z}}\,. \end{aligned} \tag{32}$$

For this particular analysis, it is assumed that the orbital angular velocity is locally constant. The objective is to determine suitable position and rate feedback gains that will increase spacecraft robustness. The nominal characteristic equation for any second order system has the following form

$$\Lambda(s) = s^2 + 2\omega_n\zeta s + \omega_n^2\,. \tag{33}$$

The natural frequency is denoted as ω_n and ζ is the damping factor, which will be chosen to be one. Each of the denominators in Equation (32) will be equated to Equation (33). Solving for the coefficients, the result is two equations and three unknowns. The third equation makes use of the final value theorem, and it is given by the following

$$f(\infty) = \lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s)\,. \tag{34}$$

The pointing requirements for this satellite require a steady state pointing accuracy of 0.1° about each axis. By applying the final value theorem to Equation (32) and if the external disturbance torques can be approximated as a step input, position feedback gains can be determined from the following equations

$$\begin{aligned} k_x &= \frac{T_x - 4\Omega^2(I_y - I_x)\phi_{ss} + \Omega h_y \phi_{ss}}{\phi_{ss}}, \\ k_y &= \frac{T_y - 3\Omega^2(I_x - I_z)\theta_{ss}}{\theta_{ss}}, \\ k_z &= \frac{T_z - \Omega^2(-I_x + I_y)\psi_{ss} + \Omega h_y \psi_{ss}}{\psi_{ss}}. \end{aligned} \tag{35}$$

The '*ss*' subscript denotes steady state and the design torques represent a worst-case scenario. It can be seen from Equation (35) that the position feedback gains are not constant; they will vary as a function of orbital position. The natural frequency for roll, pitch, and yaw can now be determined by taking the square root of the last term in the denominator. Once this is found, the velocity feedback gains can be calculated from the following expressions

$$\begin{aligned} k_{vx} &= 2\omega_{nx} I_x\,, \\ k_{vy} &= 2\omega_{ny} I_y\,, \\ k_{vz} &= 2\omega_{nz} I_z\,. \end{aligned} \tag{36}$$

In a similar manner to the position feedback gains, the velocity feedback gains also vary with time. Equation (36) and Figure 7 each depicts the time varying nature of the PD gains over one orbit, specifically during perigee. As expected, the pitch and pitch rate gains are much higher than the roll and yaw gains [7].

H. KINEMATICS

The inertial frame, orbital frame, and body frame are the three reference frames are used in the derivation of equations of motion. Direction cosine matrices (DCM) are used to transform between coordinate systems. These matrices are given by Equations (15) and (16) above. The orbital frame of reference is oriented such that the x-axis points in the direction of the velocity vector, the z-axis points towards the center of the Earth, and the y-axis completes the right-hand set. The body frame is aligned with the orbital frame, as that is the direction of motion. The following 3-2-1 transformations from the orbital frame to the body frame are given by Equation (17). The orbital frame rotates at a rate of $\Omega(t)$ with respect to the inertial reference frame, or

$$^{i}\vec{\omega}^{o} = -\Omega\hat{o}_2 . \tag{37}$$

To perform angular momentum calculations, the inertial angular velocity is expressed in body coordinates. This is represented by

$$^{i}\vec{\omega}^{b} = ^{i}\vec{\omega}^{o} + ^{o}\vec{\omega}^{b} . \tag{38}$$

Expressed in body coordinates, the angular velocity of the orbital frame with respect to the inertial frame, is

$$^{i}\vec{\omega}_{b}^{o} = -^{b}C^{o}\Omega\hat{o}_2 . \tag{39}$$

Angular velocity of the body frame with respect to the orbital frame, expressed in body coordinates, is

$$^{o}\vec{\omega}_{b}^{b} = \dot{\phi}\hat{b}_1 + C(\phi)C(\theta)\dot{\theta}\hat{n}_2 + {}^{b}C^{o}\dot{\psi}\hat{o}_3 . \tag{40}$$

The $\hat{n}_2$ unit vector belongs to an intermediate reference frame. If Equations (39) and (40) are substituted into (38), the following result is obtained

$$^{i}\vec{\omega}^{b} = (\dot{\phi} - \Omega\psi)\hat{b}_1 + (\dot{\theta} - \Omega)\hat{b}_2 + (\dot{\psi} + \Omega\phi)\hat{b}_3 . \tag{41}$$

Equation (41) is a simplified expression using small angle approximations and neglecting second order terms [6], [7], [8].

I. DERIVATION OF EQUATIONS OF MOTION

In determining the attitude of a satellite, the common approach is to translate all factors into the body coordinate system, since on-board sensors are designed to detect errors with respect to the body frame. As noted previously, Equation (34) was obtained using small angle approximations where errors are represented by: $\phi \equiv$ roll error, $\theta \equiv$ pitch error, and $\psi \equiv$ yaw error. Total spacecraft angular momentum can be separated into two angular momentum vectors for the spacecraft body and the reaction wheels, which is given by the expression

$$\vec{H} = \vec{H}_b + \vec{H}_w . \tag{42}$$

If it is assumed that cross products of inertia are negligible, then

$$\vec{H}_b = I\,^{i}\vec{\omega}^{b} . \tag{43}$$

Note that when calculating the angular momentum of the satellite about its center of mass, inertial angular rates must be used rather than body rates. Substituting Equations (41) and (42) into Equation (43), the total spacecraft angular momentum is

$$\vec{H} = \left(I_x\dot{\phi} - I_x\Omega\psi + h_x\right)\hat{b}_1 + \left(I_y\dot{\theta} - I_y\Omega + h_y\right)\hat{b}_2 + \left(I_z\dot{\psi} + I_z\Omega\phi + h_z\right)\hat{b}_3 . \tag{44}$$

Next, the relation

$$\frac{d}{dt}^{i}\vec{H} = \frac{d}{dt}^{b}\vec{H} + {}^{i}\vec{\omega}^{b} x \vec{H}\,, \tag{45}$$

is used to determine the Euler moment equations. Neglecting second order terms and gravity gradient moments, Euler equations for this spacecraft are

$$\begin{aligned} T_x &= I_x\ddot{\phi} + 4\Omega^2\left(I_y - I_z\right)\phi - \Omega h_y\phi - \Omega h_z + \Omega\left(-I_x + I_y - I_z\right)\dot{\psi} - h_y\dot{\psi} + h_z\dot{\theta} - I_x\dot{\Omega}\psi + \dot{h}_x\,, \\ T_y &= I_y\ddot{\theta} + 3\Omega^2\left(I_x - I_z\right)\theta + h_x\dot{\psi} + \Omega h_z\psi + \Omega h_x\phi - h_z\dot{\phi} - I_y\dot{\Omega} + \dot{h}_y\,, \\ T_z &= I_z\ddot{\psi} + \Omega^2\left(-I_x + I_y\right)\psi - \Omega h_y\psi + \Omega h_x + \Omega\left(I_x - I_y + I_z\right)\dot{\phi} - h_x\dot{\theta} + h_y\dot{\phi} + I_z\dot{\Omega}\phi + \dot{h}_z\,. \end{aligned} \tag{46}$$

These equations describe the motion of the spacecraft when subject to external disturbance torques. Rates of change of angular momentum of each reaction wheel will be used to counteract the disturbance moments, maintaining the required pointing accuracy. However, solving for the Euler angles is non-trivial as all three differential equations are second order and coupled. If the cross products of inertia are not negligible, the equations of motion become [6], [7], [8],

$$\begin{aligned} T_{x1} &= T_x + (\ddot{\theta} - \dot{\Omega} - 3\Omega^2\theta)I_{xy} + (\Omega^2\psi + \ddot{\psi} + \dot{\Omega}\phi)I_{xz} + (-2\Omega\dot{\theta} - 2\Omega^2)I_{yz}\,, \\ T_{y1} &= T_y + (-2\Omega\dot{\psi} + 2\Omega^2\phi + \ddot{\phi} - \dot{\Omega}\psi)I_{xy} + 3\Omega^2 I_{xz} + (2\dot{\phi}\Omega - \Omega^2\psi + \ddot{\psi} + \dot{\Omega}\phi)I_{yz}\,, \\ T_{z1} &= T_z + (2\Omega\dot{\theta} - \Omega^2)I_{xy} + (-2\Omega^2\phi + \ddot{\phi} - \dot{\Omega}\psi)I_{xz} + (\ddot{\theta} - \dot{\Omega} - 3\Omega^2\theta)I_{yz}\,. \end{aligned} \tag{47}$$

J. CONTROL LAWS

To determine control laws for the magnetic torquers, one must define what needs to be controlled. For the satellite, the best approach would be to join all the equations in a matrix form and only then linearize them. An engineer would come up with a MIMO control law that would control the system as a whole and not by parts. This approach has one important advantage: it does not matter if the state variables are coupled.

The only concern becomes the linearization. As a side effect, one can use the model of the plant in a Kalman filter configured as an observer or, if the noise sources are not relevant, use a deterministic observer. The engineer could then use the wealth of

methods that are present in the linear state space approach, especially adaptive, robust, and optimal, control techniques.

The control law proposed here is by no means the best one, only suitable for the model. Control laws were developed for pitch, by using the rate of momentum of a wheel, and for roll/yaw, by using reaction jets. In turn, both the reaction jets and the momentum wheel laws, generated torque commands. These were fed into the physical devices that would produce the actual torque. It is reasonable to argue that any physical device that can produce the toque required by the control law would be adequate. Let us keep this reasoning in mind when going over the next paragraphs.

Suppose that it was possible to generate, with the magnetic torquers, the same torque that is generated by the reaction jets. Imagine that there is a box that can accept the torque required and outputs the commands for the torquers. One could then connect the output of the reaction jets control law (i.e., a number that requests the amount of torque required) and feed it into that magic box. Now, we can bring this forward a step if the control law assumes the system is linear. Of several implications, one of the most important is part of the very definition of linearity: the superposition principle must apply. Next, we look at the pitch control. The pitch controller generates a number that represents the amount of torque the pitch wheel must generate. If we take this number and sum it up with the output by the reaction jets controller, we would have the total torque needed to control the spacecraft attitude.

If we take the total torque that is needed to control the spacecraft and feed it to the controller, the torquers could produce all the torque needed. No change in the angular velocity of the wheel would be needed, and no more momentum dumping, and no more

wheel over speed and a larger MTTF. However, open questions remain, such as; what goes inside the control algorithm, and are the torquers able to generate all the torque needed to control the spacecraft?

K. MEAN TIME TO FAILURE

1. Using the Earth's Magnetic Field

Suppose that one has the Earth magnetic field vector **B** and the amount of torque desired $\mathbf{T}_d$. We need to know the value **m** needed for the magnetic torquers that will produce $\mathbf{T}_p$ in the presence of **B**. Note that the torque produced may be, as will be seen, different from the desired one

$$\mathbf{T}_p = \mathbf{m} \times \mathbf{B} \tag{48}$$

From Equation (48) follows Equation (49). This equation will give the direction of the magnetic torquers vector. However, the engineer must be cautious: $\mathbf{T}_d$ will only be the output of the torquers when $\mathbf{T}_d \perp \mathbf{B}$. In all the other circumstances, the best torque obtainable is the portion of $\mathbf{T}_d$, which is perpendicular to **B**. That portion of $\mathbf{T}_d$ is then called $\mathbf{T}_p$

$$\hat{\mathbf{m}} = \hat{\mathbf{B}} \times \hat{\mathbf{T}}_d \,. \tag{49}$$

The amplitude of **m** can be found using Equation (50)

$$\|\mathbf{m}\| = \frac{\|\hat{\mathbf{T}}_d\|}{\|\mathbf{B}\| \dfrac{\sin\theta}{\mathbf{mB}}} = \frac{\|\hat{\mathbf{T}}_p\|}{\|\mathbf{B}\|} \,. \tag{50}$$

The engineer will then have to make a choice: is $\mathbf{T}_d$ replaced in Equation (50) with $\mathbf{T}_p$? The question can only be answered with further analysis and simulations. In this work, Equation (50) was used with a slight variation; shown in Equation (51), note that the $\mathbf{T}_p$ was changed to $\mathbf{T}_d$,

$$\|\mathbf{m}\| = \frac{\|\mathbf{T}_d\|}{\|\mathbf{B}\|}. \tag{51}$$

This law leaves a question open: what happens when the Earth's magnetic field is parallel to $\mathbf{T}_d$? Nothing, because no torque can be generated, resulting in lack of control over the spacecraft. However, as the time progress, the magnetic field moves with respect to the spacecraft and control is regained. How the shortage is expected to be remains an open question. What are the consequences of $\mathbf{T}_d \neq \mathbf{T}_p$? The answer to this question depends on more analysis and simulations. It seems that for somewhat low precision (0.1°) the difference probably can be compensated by the controller. In the simulations the steady state for small and large angles was $\mathbf{e} = (0\ 0\ 0\ 1)^T$. The results with disturbances were similar to the one produced with the reaction jets. $\mathbf{B}_b$ does have discontinuities. What happens with the control? Every time the **B** measured in the body coordinates changes its signal, the control law would need to have discontinuities also. Since the discontinuities are when the $\mathbf{T}_d$ is almost parallel to **B**, the best solution is to turn off the controller and let the spacecraft drift until this condition vanishes. This is probably the best approach because of the magnetic field sensors noise and miss alignment.

There is another concern in the actual implementation: the torquers cannot be turned on all the time. The torquers must be turned off in order to perform accurate magnetic field measurements. If the torquers are active only half of the time, then the torquers commands must be multiplied by a factor of two. It is easy to come out with the correction factors for other operation conditions.

L. DISCRETE KALMAN FILTER

The Kalman Filter is a recursive algorithm for estimating a state vector given past estimates and current measurements with noise. With a system model of the plant dynamics and sensor noise, the filter will minimize the mean square error. Since it was first development in 1960 by R. E. Kalman, the filter has been used in numerous fields of study and many sources exist that walk through the derivation of his work. For simplicity, only the resulting equations will be shown here [3], [8].

The filter itself is a two step process, a prediction followed by an update. In this simulation, a single measurement is used as the initial prediction. The simulation code therefore first computes the Kalman Gain with the initial covariance prediction before updating the covariance prediction and then making a new prediction.

1. The Process to Be Estimated

The Kalman filter addresses the general problem of trying to estimate the state $x \in \Re^{\aleph}$ of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_{k+1} = A_k x_k + Bu_k + w_k, \tag{52}$$

with a measurement $z_k \in \Re^{\aleph}$ that is

$$z_k = H_k x_k + v_k. \tag{53}$$

The random variables w_k and v_k represent the process and measurement noise respectively. They are assumed independent of each other, white, and with normal probability distributions

$$p(w) - N(0, Q),$$
$$p(v) - N(0, R). \tag{54}$$

The $n \times n$ matrix A in the difference Equation (68) relates the state at time step k to the state at step $k+1$, in the absence of either a driving function or process noise. The matrix B relates the control input to the state x. The matrix H in the measurement Equation (53) relates the state to the measurement z_k.

2. The Computational Origins of the Filter

We define $\hat{x}_k^- \in \Re^N$ (note the "super minus") to be our a priori state estimate at step k given knowledge of the process prior to step k, and $\hat{x}_k \in \Re^N$ to be our a posteriori state estimate at step k given measurement z_k. We can then define a priori and a posteriori estimate errors as

$$c_k^- \equiv x_k - \hat{x}_k^-, \tag{55}$$

and

$$c_k \equiv x_k - \hat{x}_k. \tag{56}$$

The a priori estimate error covariance is then

$$P_k^- = E\left[e_k^- e_k^{-T}\right], \tag{57}$$

and the a posteriori estimate error covariance is

$$P_k = E\left[e_k e_k^T\right]. \tag{58}$$

In deriving the equations for the Kalman filter, we begin with the goal of finding an equation that computes an a posteriori state estimate $\hat{x}_k$ as a linear combination of an a priori estimate $\hat{x}_k^-$ and a weighted difference between an actual measurement and a measurement prediction as shown below in Equation (59). Some justification for

Equation (59) is given in the sub-section "The Probabilistic Origins of the Filter," found below

$$\hat{x}_k = \hat{x}_k^- + K\left(z_k - H_k\hat{x}_k^-\right). \tag{59}$$

The difference $\left(z_k - H_k\hat{x}_k^-\right)$ in Equation (59) is called the measurement innovation, or the residual. The residual reflects the discrepancy between the predicted measurement $H_k\hat{x}_k^-$ and the actual measurement z_k. A residual of zero means that the two are in complete agreement.

The $n \times m$ matrix *K* in Equation (59) is chosen to be the gain or blending factor that minimizes the a posteriori error covariance, Equation (58). This minimization can be accomplished by first substituting Equation (59) into the above definition for e_k, substituting that into Equation (58), performing the indicated expectations, taking the derivative of the trace of the result with respect to *K*, setting that result equal to zero, and then solving for *K*. One form of the resulting *K* that minimizes Equation (59) is given by

$$\begin{aligned} K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k\right)^{-1} \\ &= \frac{P_k^- H_k^T}{H_k P_k^- H_k^T + R_k} \end{aligned} . \tag{60}$$

Looking at Equation 60, we see that as the measurement error covariance R_k approaches zero, the gain *K* weights the residual more heavily. Specifically

$$\lim_{R_k \to 0} K_k = H_k^{-1}. \tag{61}$$

On the other hand, as the a priori estimate error covariance R_k approaches zero, the gain *K* weights the residual less heavily. Specifically

$$\lim_{P_k^x \to 0} K_k = 0. \tag{62}$$

Another way of thinking about the weighting by K is that as the measurement error covariance R_k approaches zero, the actual measurement z_k is "trusted" increasingly, while the predicted measurement $H_k\hat{x}_k^-$ is trusted increasingly less. On the other hand, as the a priori error covariance estimate P_k^- approaches zero, the actual measurement z_k is also trusted increasingly less, while the predicted measurement $H_k\hat{x}_k^-$ is increasingly trusted.

3. The Probabilistic Origins of the Filter

The justification for Equation (59) is rooted in the probability of the a priori estimate $\hat{x}_k^-$ conditioned on all prior measurements z_k (Bayes' rule). For now let it suffice to point out that the Kalman filter maintains the first two moments of the state distribution

$$\begin{aligned} E[x_k] &= \hat{x}_k \\ E\left[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\right] &= P_k \end{aligned}. \tag{63}$$

The a posteriori state estimate equation 60 reflects the mean (the first moment) of the state distribution—it is normally distributed if the conditions of Equations (54) are met. The a posteriori error covariance estimate, Equation (58), reflects the variance of the state distribution (i.e., the second non-central moment). In other words

$$P(x_k \mid z_k) - N\left(E[x_k], E\left[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\right]\right) = N(\hat{x}_k, P_k). \tag{64}$$

4. The Discrete Kalman Filter Algorithm

We begin this section with a broad overview covering the "high-level" operation of one form of the discrete Kalman filter. After presenting this high-level view, we will narrow the focus to the specific equations and their use in this version of the filter.

The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, the equations for the Kalman filter fall into two groups: 1) time update equations and, 2) measurement update equations. The time update equations are responsible for projecting forward in time the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for the feedback, (i.e.) for incorporating a new measurement into the a priori estimate to obtain an improved a posteriori estimate.

The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations. Indeed the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems as shown below in Figure 5.

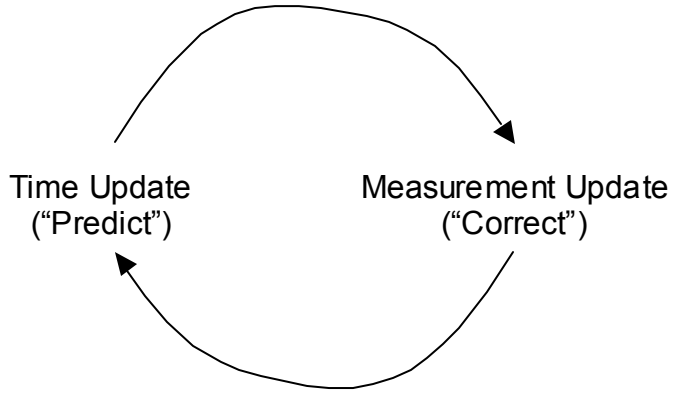


Figure 5: The ongoing discrete Kalman filter cycle.

The time update projects the current state estimate ahead in time, while the measurement update adjusts the projected estimate by an actual measurement at that time.

The specific equations for the time updates are presented below in Equations (65) while the measurement updates are presented below in Equations (66).

$$\hat{x}_{k+1} = A_k \hat{x}_k + Bu_k ,$$
$$P_{k+1}^- = A_k P_k A_k^T + Q_k . \tag{65}$$

Notice how the time update in Equations (65) project the state and covariance estimates from time step *k* to step *k+1*. A_k and *B* are from Equation (52), while Q_k is from Equation (54). Initial conditions for the filter were discussed in earlier references

$$\begin{aligned} K_k &= P_k^- H_k^T \left(H_k P_k^- H_k^T + R_k \right)^{-1}, \\ \hat{x}_k &= \hat{x}_k^- + K\left(z_k - H_k \hat{x}_k^- \right), \\ P_k &= \left(I - K_k H_k \right) P_k^- . \end{aligned} \tag{66}$$

The first task during the measurement update is to compute the Kalman gain, K_k. Note that the equation given here, as Equation (66), is the same as Equation (60). The next step is to actually measure the process to obtain z_k, and then to generate an a posteriori state estimate by incorporating the measurement as in Equation (66). Again, Equation (66) is simply Equation (59) repeated here for completeness. The final step is to obtain an a posteriori error covariance estimate via Equation (66).

After each time and measurement update pair, the process is repeated with the previous a posteriori estimates used to project or predict the new a priori estimates. This recursive nature is one of the very appealing features of the Kalman filter as it makes practical implementations much more feasible than, say for example, an implementation of a Weiner filter which is designed to operate on all of the data directly for each estimate. Instead, the Kalman filter recursively conditions the current estimate on all of the past measurements. Figure 6 below offers a complete picture of the operation of the filter, combining the high-level diagram of Figure 5 with the equations from Equations (65) and Equations (66).

5. Filter Parameters and Tuning

In the actual implementation of the filter, each of the error measurement covariance matrices R_k and the process noise Q_k, as given by Equations (54), might be measured prior to operation of the filter. In the case of the measurement error covariance R_k, in particular, this makes sense-because we need to be able to measure the process, while operating the filter; generally, we should be able to take some off-line sample measurements in order to determine the variance of the measurement error.

In the case of Q_k, oftentimes the choice is less deterministic. For example, this noise source is often used to represent the uncertainty in the process model shown in Equation (52). Sometimes a very poor model can be used simply by "injecting" enough uncertainty via the selection of Q_k values. In this case, one would hope that measurements of the process would be reliable.

In either case, whether or not we have a rational basis for choosing the parameters, statistically speaking, superior filter performance can be obtained by "tuning" the filter parameters Q_k and R_k. The tuning is usually performed off-line, frequently with the help of another distinct Kalman filter.

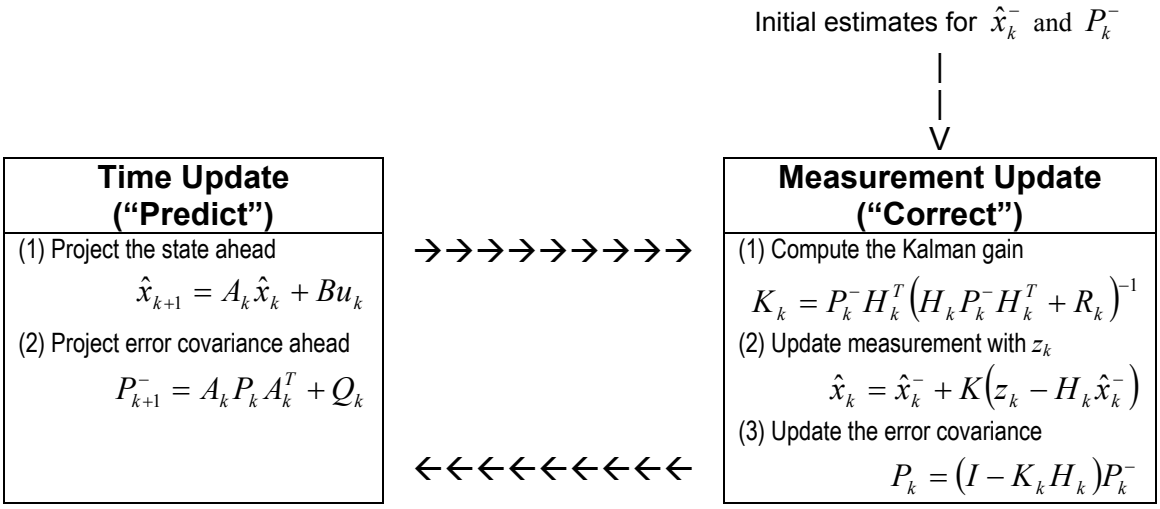


Figure 6: A complete picture of the operation of the Kalman filter, combining the high-level diagram of Figure 5 with the equations from Equations (65) and (66).

In closing we note that under conditions where Q_k and R_k are constant, both the estimation error covariance P_k and the Kalman gain K_k will stabilize quickly and then remain constant (see the filter update equations in Figure 6). If this is the case, these parameters can be pre-computed by either running the filter off-line or, for example, by determining the steady-state P_k value.

It is frequently the case however that the measurement error does not remain constant. For example, when sighting beacons in our optoelectronic tracker ceiling panels, the noise in measurements of nearby beacons will be smaller than in beacons that are more distant. In addition, the process noise Q_k is sometimes changed dynamically during filter operation in order to adjust to different dynamics. As an example, in the case of tracking the head of a user of a 3D virtual environment we might reduce the magnitude of Q_k if the user seems to be moving slowly, and increase the magnitude if the dynamics start changing rapidly. In such a case Q_k can be used to model not only the uncertainty in the model, but also the uncertainty of the user's intentions.

A 6-state discrete Kalman filter has been chosen to estimate both position and rates from noisy star sensor data. The Kalman filter that will be used in the simulation is represented by

$$\vec{x}_{k+1} = \Phi_k \vec{x}_k + \Delta_k \vec{u}_k + \vec{w}_k \,, \qquad \vec{z}_k = H\vec{x}_k + \vec{v}_k \,. \tag{67}$$

The white sequence w_k for the plant has a covariance, Q, while the sensor's noise v_k has a covariance, R. Noise from the star sensor is affected by the magnitude of the star; a bright star is noisier than a dim star. The sensor noise covariance is defined as follows

$$R_k = E\left[\vec{v}_k \vec{v}_k^T\right]. \tag{68}$$

M. DERIVATION OF THE Q MATRIX

Solving for the covariance of the plant noise is no trivial matter. In this simulation, the Q matrix will vary with each time step. The formal definition of the plant noise covariance is given by [3], [10]

$$Q_k = E\left[\vec{w}_k \vec{w}_k^T\right] \tag{69}$$

It can be shown that Equation (69) must satisfy the following matrix differential equation

$$\dot{Q}_k = A_{aug} Q_k + Q_k A_{aug}^T + BWB^T \,. \tag{70}$$

The augmented A_{aug} matrix is defined from $\dot{\vec{x}} = (A - BF)x + Bu_d$ as the quantity, *A-BF,* and the power spectral density matrix associated with the forcing function $\vec{u}$ is denoted by W.

The solution to Equation (70) is greatly simplified for the time invariant case. It proceeds as follows,

$$\alpha = \begin{bmatrix} -A_{aug} & BWB^T \\ 0 & B^T \end{bmatrix} \Delta t \,. \tag{71}$$

By taking the matrix exponential of Equation (71), the following result is obtained

$$\beta = \begin{bmatrix} \cdots & \chi^{-1} Q_k \\ 0 & \chi^T \end{bmatrix}. \tag{72}$$

The upper left partition can be neglected for this analysis. The plant noise covariance matrix can now be determined by multiplying the upper right partition of Equation (72) by χ. This method was first formulated by Van Loan in 1978.

N. KALMAN ALGORITHM

Before entering the Kalman filter loop, an initial estimate $\hat{x}_0^-$, and its error covariance P_0^-, must chosen. The '-' superscript will represent the predicted estimate while the '^' notation denotes estimation. The discrete Kalman filter is, in essence, just a computer algorithm that derives optimal estimates from discrete measurements. Although there are different forms of the discrete Kalman filter, the most fundamental form starts with the Kalman gain calculation, which is given by [3], [10]

$$G_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \,. \tag{73}$$

The value of this gain matrix will vary with each time step. Next, it is required to update the estimate using star sensor data to determine the accuracy of this new estimate. These equations are given as

$$\begin{aligned} \hat{\bar{x}}_k &= \hat{\bar{x}}_k^- + G_k \left(\bar{z}_k - H_k \hat{\bar{x}}_k^- \right), \\ P_k &= \left(I - G_k H_k \right) P_k^- \,. \end{aligned} \tag{74}$$

Equation (75), shown below, illustrates the recursive nature of the discrete Kalman filter. A favorable characteristic of any recursive filter is that there is no need to store past measurements.

Equation (74) is the actual output of the discrete Kalman filter. It estimates both attitude angles and attitude rates given only star sensor angle information. Not only does it derive rates, but it also mitigates sensor noise effects. Lastly, it is necessary to project ahead and estimate the state for the next time step. The predictive equations are as follows

$$\begin{aligned} \hat{\bar{x}}_{k+1} &= \Phi_k \hat{\bar{x}}_k + \Delta_k \vec{u}_k \,, \\ P_{k+1}^{-} &= \Phi_k P_k \Phi_k^T + Q_k \,. \end{aligned} \tag{75}$$

It is interesting to note that in Equation (75), the deterministic forcing function has been included. This forcing function consists of known reaction wheel moments, which can be measured by the reaction wheel motor assembly. If this deterministic term is not included, the rate estimator is unable to accurately estimate satellite rates near perigee.

For the purpose of analysis and proper tuning, it is helpful to look at the time-varying nature of both the Q and P matrices over a period of one orbit. Since the off-diagonal elements of these matrices are small, only the diagonal elements will be examined. These elements are shown in Figure 16.

O. STATE SPACE EQUATIONS OF MOTION

The equations of motion completely describe the motion of the satellite when subjected to both internal and external disturbance moments. If the body accelerations are solved for in Equation (47), the following result is obtained [8], [10]

$$\ddot{\phi} = \left[\frac{-4\Omega^2\left(I_y - I_z\right)\phi + \Omega h_y \phi - \Omega\left(-I_x + I_y - I_z\right)\psi + h_y\dot{\psi} - h_z\dot{\theta} + I_x\dot{\Omega}\psi}{I_x}\right] - \left[\frac{\dot{h}_x}{I_x}\right] + \left[\frac{T_{spx} + \Omega h_z}{I_x}\right],$$

$$\ddot{\theta} = \left[\frac{-3\Omega^2\left(I_x - I_z\right)\theta - h_x\dot{\psi} - \Omega h_z\psi - \Omega h_x\phi + h_z\dot{\phi}}{I_y}\right] - \left[\frac{\dot{h}_y}{I_y}\right] + \left[\frac{T_{spy} + I_y\dot{\Omega}}{I_y}\right], \tag{76}$$

$$\ddot{\psi} = \left[\frac{-\Omega^2\left(-I_x + I_y\right)\psi + \Omega h_y\psi - \Omega\left(I_x - I_y + I_z\right)\dot{\phi} + h_x\dot{\theta} - h_y\dot{\phi} - I_z\dot{\Omega}\phi}{I_z}\right] - \left[\frac{\dot{h}_z}{I_z}\right] + \left[\frac{T_{spz} - \Omega h_x}{I_z}\right].$$

For computational reasons, it is desirable to reduce these second order equations to first order equations by making the following state variable substitutions

$$\vec{x} = [\phi \quad \dot{\phi} \quad \theta \quad \dot{\theta} \quad \psi \quad \dot{\psi}]^T. \tag{77}$$

With these definitions, we can translate the satellite dynamic equations into the following matrix form

$$\dot{\vec{x}} = A\vec{x} + B\vec{u}. \tag{78}$$

A is the plant matrix and it is given by

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-4\Omega^2(I_y - I_z) + \Omega h_y}{I_x} & 0 & 0 & \frac{-h_z}{I_x} & \dot{\Omega} & \frac{-\Omega(-I_x + I_y - I_z) + h_y}{I_x} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{-\Omega h_y}{I_y} & \frac{-h_z}{I_y} & \frac{-3\Omega^2(I_x - I_z)}{I_y} & 0 & \frac{-\Omega h_z}{I_y} & \frac{-h_x}{I_y} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\dot{\Omega} & \frac{-\Omega(I_x - I_y + I_z) - h_y}{I_z} & 0 & \frac{h_x}{I_z} & \frac{-\Omega^2(-I_x + I_y) + \Omega h_y}{I_z} & 0 \end{bmatrix}. \tag{79}$$

B is the control matrix given by

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{80}$$

u is going to be the input reaction wheel control, and will have the following form

$$u = -Fx + u_d. \tag{81}$$

F will be the PD controller gain matrix and u_d will represent the summation of the solar pressure moments and the internal reaction wheel moments. F and u_d are given by

$$F = \begin{bmatrix} \frac{k_x}{I_x} & \frac{k_{vx}}{I_x} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{k_y}{I_y} & \frac{k_{vy}}{I_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{k_z}{I_z} & \frac{k_{vz}}{I_z} \end{bmatrix},$$

$$u_d = \begin{bmatrix} \frac{T_{spx} + \Omega h_z}{I_x} \\ \frac{T_{spy} + I_y \dot{\Omega}}{I_y} \\ \frac{T_{spz} - \Omega h_x}{I_z} \end{bmatrix}. \tag{82}$$

Substituting Equation (82) into Equation (78), the following equation of motion is obtained

$$\dot{\tilde{x}} = (A - BF)x + Bu_d \,. \tag{83}$$

Equation (83) is equivalent to the equations of motion.

P. THE MODELING APPROACH

The derivations excerpted in this section were employed as part of the foundation for the simulations in this project and were originally compiled in a paper by Henry D. Travis at the Naval Postgraduate School. For the sake of completeness, the full reference for this compilation is: Travis, Henry D., "Attitude Determination Using Star Tracker Data with Kalman Filters," Thesis, Naval Postgraduate School, December 2001.

1. Astrodynamics

a) Equations of Motion

In this case, the satellite is traveling along a Molniya orbit in the plane described by the radius and velocity vectors. The spacecraft's position in the orbit is defined by its

distance in kilometers from the center of the earth, r, and the true anomaly, ν, which is measured from perigee in radians. Using the basic equations of motion

$$\begin{aligned} F_r &= m(\ddot{r} - r\dot{\nu}^2), \\ F_\nu &= m(r\ddot{\nu} - 2\dot{r}\dot{\nu}), \end{aligned} \tag{84}$$

the forces in both the radial and tangential direction can be determined. Since the mass of the spacecraft is considered a point mass, the force per unit mass can be written as

$$F_r / m = -\mu / r^2 . \tag{85}$$

Because of the nature of a two-body problem the only force acting on the point mass is in the radial direction, the angular force F is zero. This leaves

$$\begin{aligned} \ddot{r} &= r\dot{\nu}^2 - \mu / r^2, \\ \ddot{\nu} &= -2\dot{r}\dot{\nu} / r . \end{aligned} \tag{86}$$

b) Modeling Molniya Orbit

We first model the system in the form

$$\begin{aligned} \dot{x} &= Ax + Bu, \\ y &= Cx + Du, \end{aligned} \tag{87}$$

by defining the state vector, x, as

$$x \equiv [\, r \; \dot{r} \; \nu \; \dot{\nu} \,]^T , \tag{88}$$

and the control vector, u, from Equations (86) as

$$u \equiv \begin{bmatrix} r\dot{\nu}^2 - \mu / r^2 \\ -2\dot{r}\nu / r \end{bmatrix}. \tag{89}$$

With all non-linearity included in the control laws, the system coefficients can be easily written as

$$A_{orbit} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$B_{orbit} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{90}$$

$$C_{orbit} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$D_{orbit} = \begin{bmatrix} 0 \end{bmatrix}.$$

With the linear system coefficient matrices defined, the next step is to compute the state transition matrix, Φ, and the convolution matrix, Δ, using the 'c2d' function in MATLAB. Thus, the entire Molniya orbit can be described using the discrete equation

$$x_{k+1} = \Phi x_k + \Delta u_k . \tag{91}$$

This orbital information is stored for future reference with the pitch controller.

2. The Discrete Kalman Filter

a) Definitions

The Kalman Filter is a recursive algorithm for estimating a state vector given past estimates and current measurements with noise. With a system model of the plant dynamics and sensor noise, the filter will minimize the mean square error. Since it was first development in 1960 by R.E. Kalman, the filter has been used in numerous fields of study and many sources exist that walk through the derivation of his work. For simplicity, only the resulting equations will be shown here. Following modern convention, the following definitions and notation will be used:

$\hat{x}_{k|k-1} \equiv$ State vector estimate at time k given measurements up to time k-1

$\hat{x}_{k|k} \equiv$ State vector estimate at time k given measurements up to time k

$e = x - \hat{x}_{k|k} \equiv$ State estimate error

$P_{k|k-1} \equiv$ Prediction of the covariance of the state vector

$P_{k|k} \equiv$ Update of the covariance of the state vector

$G \equiv$ Kalman gain

$Q \equiv$ Plant covariance matrix

$R \equiv$ Measurement covariance matrix

$\Phi \equiv$ State transition matrix

$H \equiv$ Observation matrix

$w \equiv$ Zero mean white Gaussian plant noise

$v \equiv$ Zero mean white Gaussian measurement noise

$z \equiv$ Noisy measurement

b) System Model

Before we can build the filter, a system must be developed that will adequately describe the behavior of the spacecraft. For simplicity, we have linearized the attitude dynamics equations of motion and used the small angle approximation. Looking at the equations of angular velocity in a rotating frame with a $\psi \to \theta \to \phi$ transformation

$$\omega_{BI} = \omega_{BR} + \omega_{RIB}, \tag{92}$$

where

$$\omega_{BR} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi}\sin(\theta) \\ \dot{\theta}\cos(\phi) + \dot{\psi}\cos(\theta)\sin(\phi) \\ \dot{\psi}\cos(\phi)\cos(\theta) - \dot{\theta}\sin(\phi) \end{bmatrix}, \tag{93}$$

which for small angles becomes

$$\omega_{BR} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix}, \tag{94}$$

and

$$\omega_{RIB} = \begin{bmatrix} 1 & \psi & -\theta \\ -\psi & 1 & \phi \\ \theta & -\phi & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -\omega_o \\ 0 \end{bmatrix} = \begin{bmatrix} -\psi\omega_o \\ -\omega_o \\ \phi\omega_o \end{bmatrix}, \tag{95}$$

therefore

$$\omega_{BI} = \begin{bmatrix} \dot{\phi} - \psi\omega_o \\ \dot{\theta} - \omega_o \\ \dot{\psi} + \phi\omega_o \end{bmatrix} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}. \tag{96}$$

Thus, the time rate of change of ω follows as

$$\begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} \ddot{\phi} - \dot{\psi}\omega_o \\ \ddot{\theta} - \omega_o \\ \ddot{\psi} + \dot{\phi}\omega_o \end{bmatrix}. \tag{97}$$

Using the preceding dynamics equations, we further assume that the second time derivatives of the angles are small enough to be ignored. This gives the following linear, constant coefficient matrices for the first system, which deals with only roll and yaw.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\omega \\ 0 & 0 & 0 & 1 \\ 0 & \omega & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}, \tag{98}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

$$D = [0].$$

The final step is to compute the state transition matrix, Φ and the convolution integral, Δ using the MATLAB command

$$[\ \Phi\ ,\ \Delta\ ] = \text{c2d(A,B,dt);} \tag{99}$$

which are used to promulgate the state vector using the equation

$$x_{k+1} = \Phi x_k + \Delta u_k\ . \tag{100}$$

c) Controllability

This linear time-invariant system is considered controllable if an input, *u(t)* will transfer the initial state of the system **x**(0) to the origin, **x**(t_f)=0 with t_f finite. Setting the state equation from the previous section to zero, it can be shown that the system is controllable if it satisfies the condition that the controllability matrix,

$$C_m = [B \quad AB], \tag{101}$$

has an inverse. Computing C_m for our system, it is seen that this condition is satisfied.

d) Control Gains

Two methods are presented for computing the control gain. The first is the pole placement method, and the second is a Linear Quadratic Regulator method, or LQR.

Pole placement method has been relied on for decades as a means of ensuring the poles of the closed-loop system are at desirable locations. Ackermann developed a procedure for computing the control gain and for single input systems; this algorithm is performed using the 'acker' command in MATLAB. However, for a multiple input case, the MATLAB 'place' command is used instead. The inputs for the place command are the state transition matrix, the convolution matrix, and the desired eigenvalues.

Using a Linear-quadratic regulator design for discrete-time systems is another method of computing the control gain. The gains from this method are considered optimal since the state-feedback law *u[n] = -kx[n]* minimizes the cost function

$$J = \sum \left(x' Qx + u' Ru + 2x' Nu \right), \tag{102}$$

subject to the state dynamics

$$x[n+1] = \Phi x[n] + \Delta u[n]. \tag{103}$$

The matrix **N** represents a relationship between the system noise, Q, and the measurement noise, R, and is set to zero for our system. Also returned are the Riccati equation solution and the closed-loop eigenvalues.

e) Filter Model

We begin developing the Kalman Filter by modeling the random process

$$x_{k+1} = \Phi_k x_k + w_k, \tag{104}$$

The process is observed at discrete points in time by the following relation

$$z_k = H_k x_k + v_k , \tag{105}$$

As defined earlier, the covariance matrices Q and R are given by

$$\begin{aligned} E[w_k w_i^T] &= Q_k , \\ E[v_k v_i^T] &= R_k . \end{aligned} \tag{106}$$

And the plant covariance is derived from the errors in position. Assuming a constant acceleration over one time step, *dt*, using Newton's force equations

$$\begin{aligned} x &= x_o + at^2 / 2 , \\ \dot{x} &= \dot{x}_o + at , \end{aligned} \tag{107}$$

results in an error of $\begin{bmatrix} dt2/2 & dt \end{bmatrix}$, which is squared and then entered into the covariance matrix

$$Q = q^2 \times \begin{bmatrix} dt^4/4 & dt^3/2 & 0 & 0 \\ dt^3/2 & dt^2 & 0 & 0 \\ 0 & 0 & dt^4/4 & dt^3/2 \\ 0 & 0 & dt^3/2 & dt^2 \end{bmatrix}, \tag{108}$$

and the measurement covariance is the square of the star tracker error

$$R = \begin{bmatrix} ste^2 & 0 \\ 0 & ste^2 \end{bmatrix}. \tag{109}$$

Since pitch is decoupled from roll and yaw, we omit pitch and pitch rate from the first state vector. The state vector, x, is then

$$x \equiv [\phi \dot{\phi} \psi \dot{\psi}] , \tag{110}$$

A second filter is therefore necessary for pitch and the corresponding state vector is defined here as

$$x_p \equiv [\theta \dot{\theta}] , \tag{111}$$

Similarly, the plant covariance follows as

$$Q_p = q_p^2 \times \begin{bmatrix} dt^4/4 & dt^3/2 \\ dt^3/2 & dt^2 \end{bmatrix}, \tag{112}$$

and the measurement covariance is reduced to

$$R = \left[ste^2\right]. \tag{113}$$

f) Algorithm

The filter itself is a two-step process, a prediction followed by an update. In this simulation, a single measurement is used as the initial prediction. The code therefore first computes the Kalman Gain with the initial covariance prediction before updating the covariance prediction and then making a new prediction

$$\begin{aligned} G &= P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1}, \\ P_{k|k} &= (I - GH)P_{k|k-1}, \\ P_{k|k-1} &= \Phi P_{k|k}\Phi^T + Q. \end{aligned} \tag{114}$$

Similarly, the initial state vector is first updated with the new Kalman Gain before a new prediction of the state vector is made based on the measurement residual

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + G(z - z_{k|k-1}). \tag{115}$$

The control, u, is then determined using the Optimal Control Law

$$u = -k(x - \hat{x}_{k|k-1}), \tag{116}$$

before updating the state vector and predicting the next state estimate using the state transition matrix, Φ, and convolution integral, Δ,

$$\begin{aligned} x_{k+1} &= \Phi x + \Delta u, \\ \hat{x}_{k|k-1} &= \Phi\hat{x}_{k|k} + \Delta u. \end{aligned} \tag{117}$$

The same procedure is followed to update and predict the pitch and pitch rate estimates with the new state vector, x_p.

There are several ways to initialize the Kalman Filter, including using an assumed state, a measurement, or a batch-processed state. An assumed state would be used either when the state is generally known without measurement or when the error is permitted to be large because there is sufficient time for the larger transient. A measurement approach is used when the error is expected to be small to begin with. A batch processed method would be used when the dynamics of the system would cause large changes from one measurement to the next. The expected changes will dictate the number of measurement processed to initialize the filter. Due to the accuracy of the star trackers and assumed low rate of change of the Euler Angles, the Kalman Filter could use the first measurement of the star tracker to initialize the filter. To measure the responsiveness of our filter, we will assume a zero state initially.

g) Modifications

The Kalman Filter does an excellent job of seeing through the noise to provide reliable state estimates. By taking a closer look at the filter, we see some variables that can be adjusted. Most of these adjustments will involve the system covariance matrix. In covariance manipulation, system covariance is an assumption of how much noise exists in the system. Small covariance correlates to a small amount of noise and will result in a better estimate. If the system is subjected to a large noise from an unexpected source, the filter will be unable to track the transient and the estimate will deteriorate. The desired covariance would be the smallest possible while still tracking any expected transient. Different variables and methods of adjusting covariance are:

reset threshold, discretized residual bias, alpha-beta, star gap response, and glitch/bias rejection.

IV. RESULTS

A. NPSAT-1 SIMULATION RESULTS

Results of the NPSAT-1 project described in this work, are presented in this section. No noise analysis was made due to the amount of effort required. The actual control law is very dependent upon the cross product of the magnetic field vector and the amount of torque desired. Therefore, noise in the angle between both vectors is going to have an important effect on the results. The evaluation of this control law will not be complete without a study of the noise effect. Additionally, two different results are presented here: with magnetic pitch control and without magnetic pitch control.

1. Small Angles

In this subsection, the results of the system with small initial angles and no noise are presented. The system was initialized with the quaternion set of

$$\begin{pmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{pmatrix} = \begin{pmatrix} 0.05 \\ 0.05 \\ 0.05 \\ \sqrt{1-3(0.05^2)} \end{pmatrix}. \tag{118}$$

The results are presented in the following graphs. Figures 7, 8, 9, and 10 display simulated results *with no disturbance*

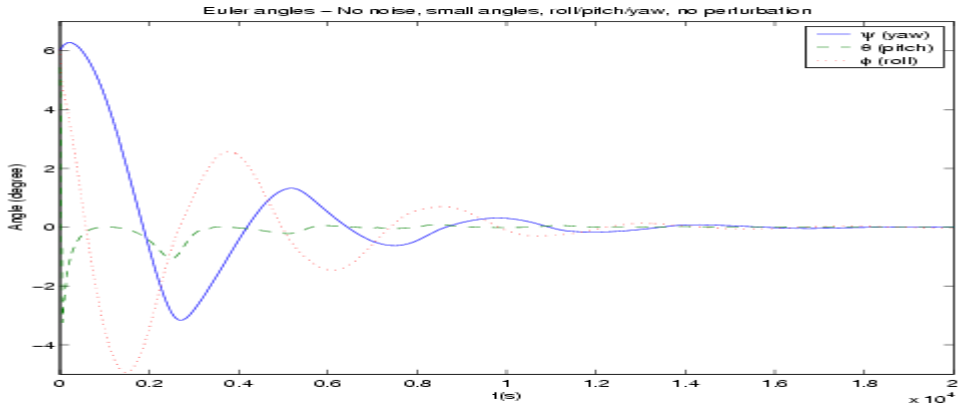


Figure 7: Small angles, mechanic pitch, with perturbation.

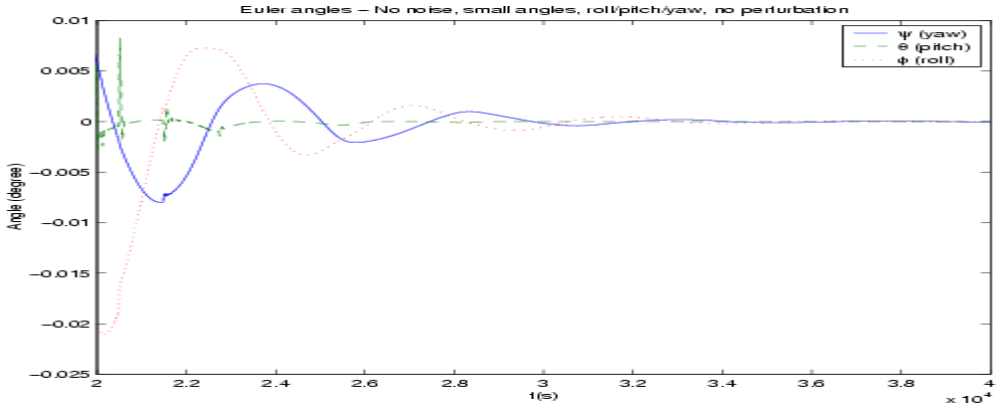


Figure 8: Small angles, mechanic pitch, with perturbation, expanded.

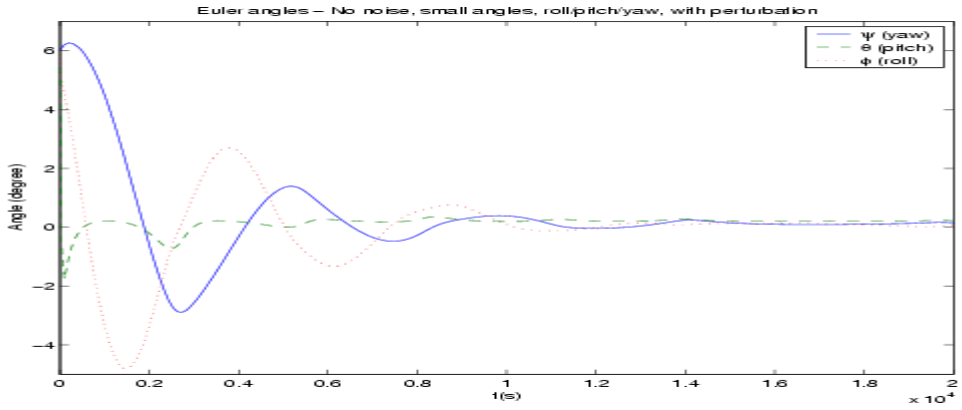


Figure 9: Small angles, mechanic pitch, with perturbation, transient response.

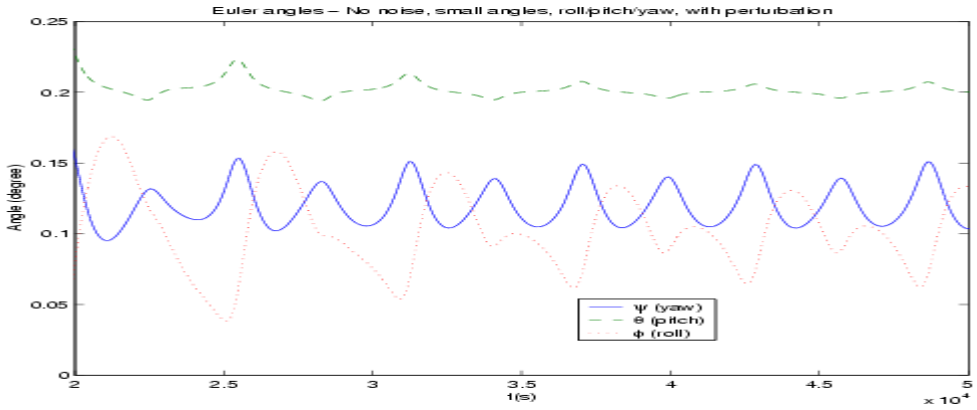


Figure 10: Small angles, mechanic pitch, with perturbation, steady state response.

Figures 11, 12, 13, and 14 display simulated results *with disturbance*:

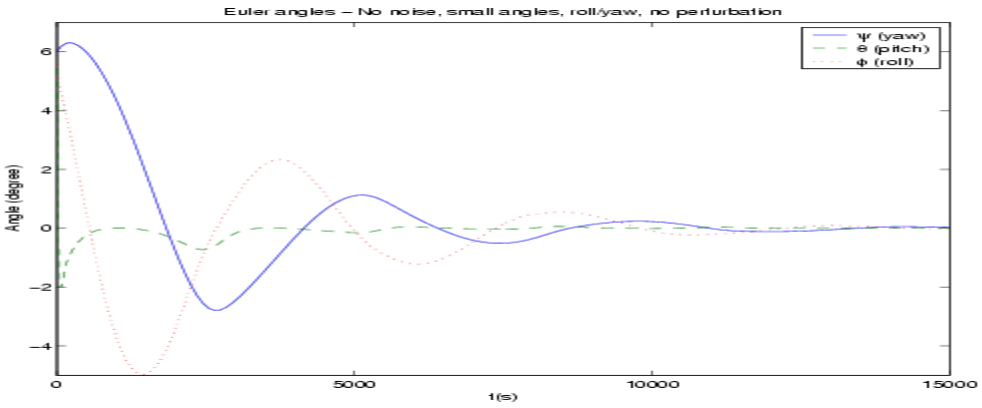


Figure 11: Small angles, mechanic pitch, with perturbation, transient response.

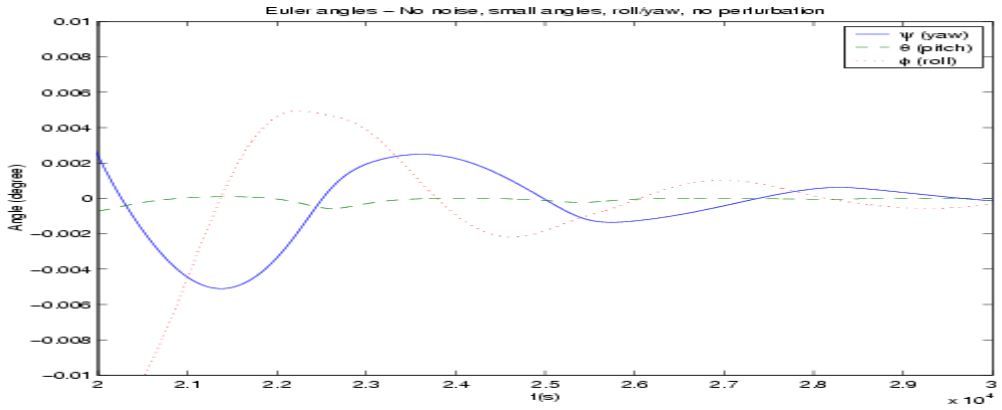


Figure 12: Small angles, mechanic pitch, with perturbation, transient response.

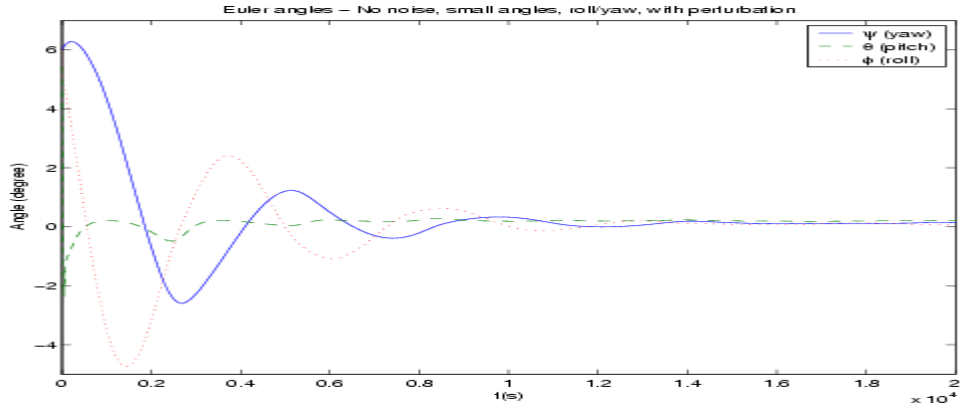


Figure 13: Small angles, mechanic pitch, with perturbation, transient response.

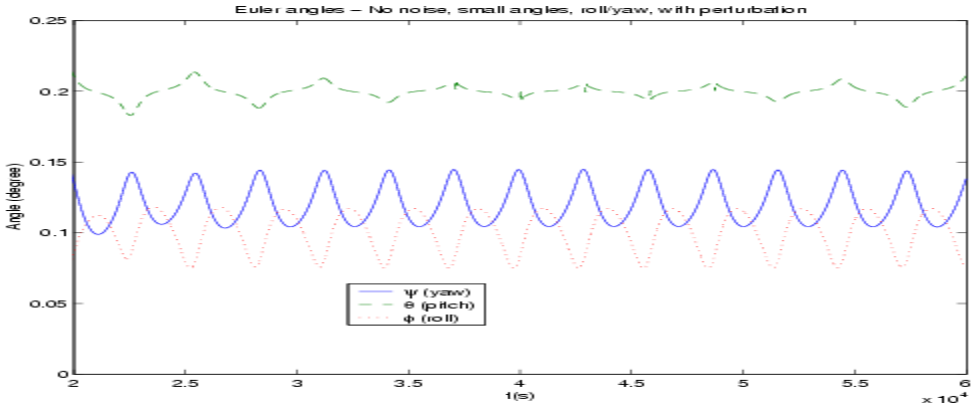


Figure 14: Small angles, mechanic pitch, with perturbation, steady state response.

2. Large Angles

In this subsection, the results of the system with small initial angles and no noise are presented. The system was initialized with the quaternion set of

$$\begin{pmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \\ \sqrt{1-3(0.5^2)} \end{pmatrix}. \tag{119}$$

Since the most interesting results for the large angles acquisition is when the system is under torque perturbations, only these results are presented here. In addition, the steady state is not relevant for this analysis, since the results are the same as with the small angle acquisition. The results of the simulation are presented as follows:

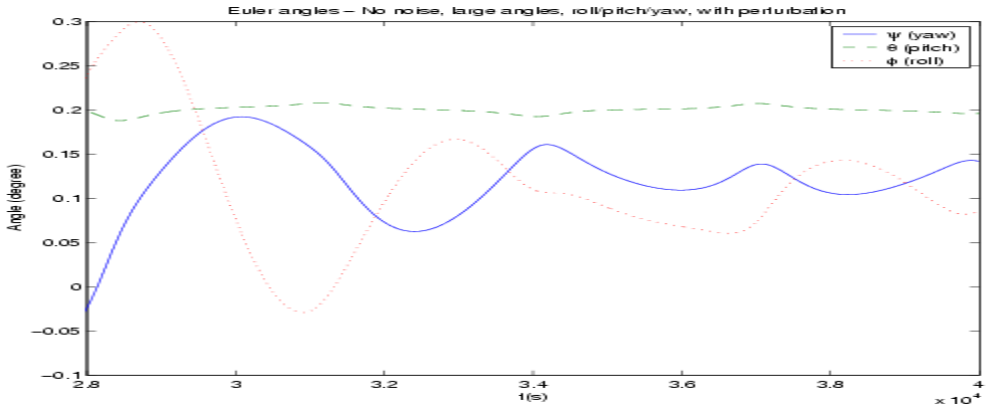


Figure 15: Large angles, magnetic pitch, with perturbation.

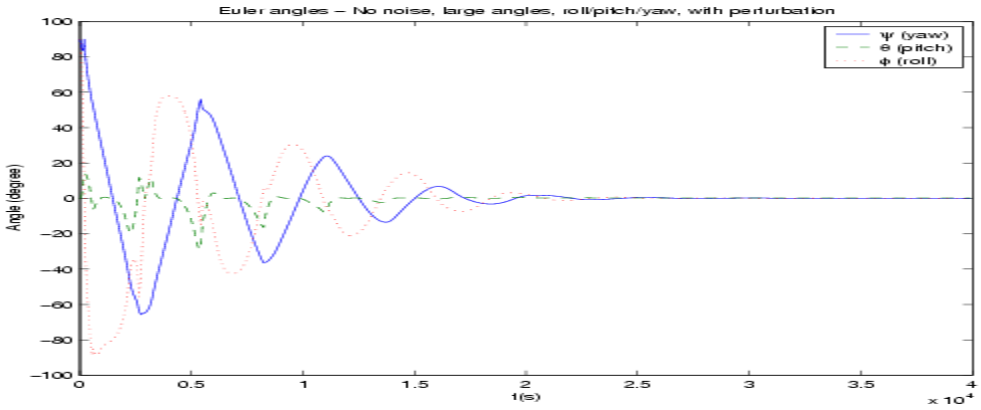


Figure 16: Large angles, mechanic pitch, with perturbation.

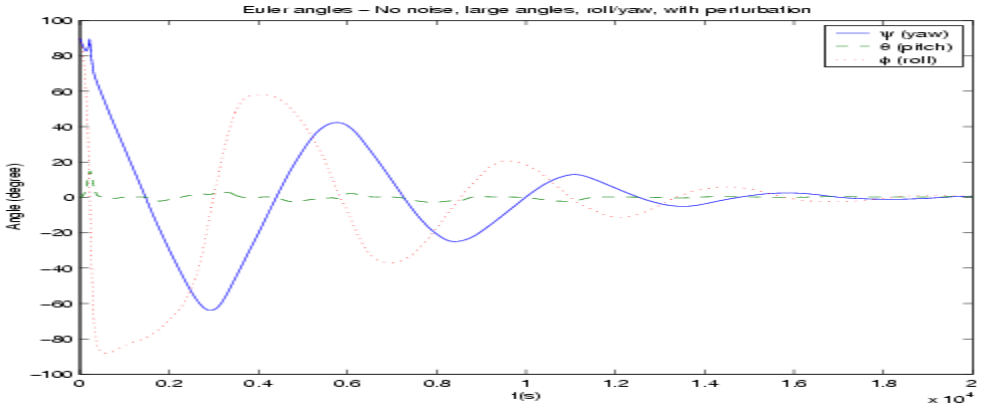


Figure 17: Large angles, mechanic pitch, with perturbation.

3. Wheel Bias Failure

An interesting result would be if the satellite could be controlled without the momentum bias wheel. Since the magnetic torquers can control roll, pitch, and yaw this would be a nice result.

The attempt of removing the wheel from the loop resulted in instability in yaw. This was expected, since the roll/yaw gains were computed for the system *with* the wheel. Interesting enough, the controller could hold very well in roll and pitch (even much better

than with the wheel on). It is believed that the controller coefficients can be recalculated to yield a stable system.

This approach is interesting in case of wheel failure. However, it is not recommended for general control. Since the removal of the wheel also removes stiffness form the system, any small torques can cause large drift on the satellite attitude in small amount of time. If that occurs, the magnetic torquers may not be able to control the attitude in a satisfactory manner due to the control shortages" created when the desired torque is aligned with the Earth magnetic field.

B. RATE ESTIMATOR SIMULATION RESULTS

The results of the 3-axis star sensor Kalman filter rate estimator are outlined below. These simulations prove that a single star sensor can provide the rate estimator required to provide additional vectors for the NPSAT-1 initial attitude determination. For the Molniya orbit case, the single axis star sensor provides estimated yaw rates based on the Kalman filter, which can be coupled with other sensors in future designs to provide a more accurate rate estimate, which in turn, will be utilized in the event of a rate gyro failure.

The results of the Kalman filter rate estimator simulations are outlined as follows:

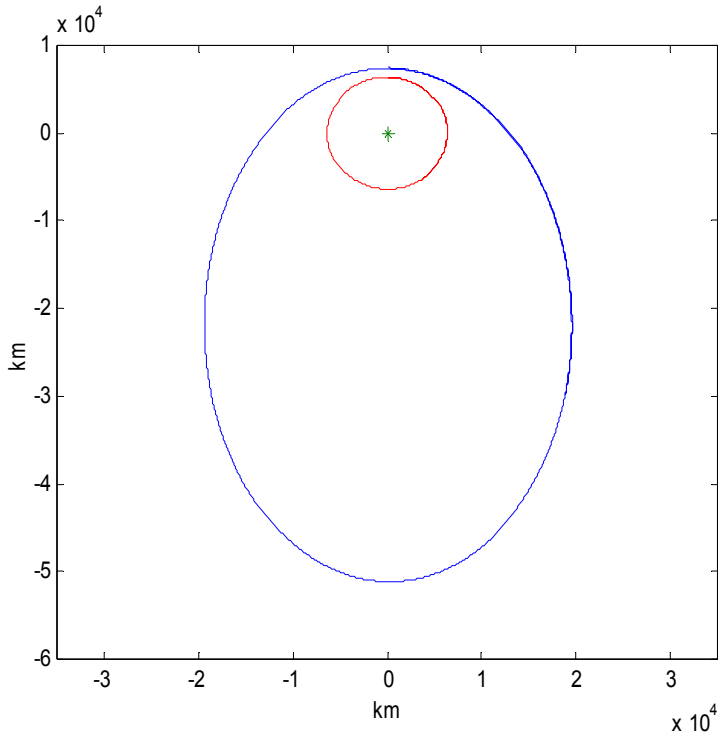


Figure 18: Molniya orbit simulation.

Figures (19), (20), and (21) represent constant gain Kalman filter with q=0.01.

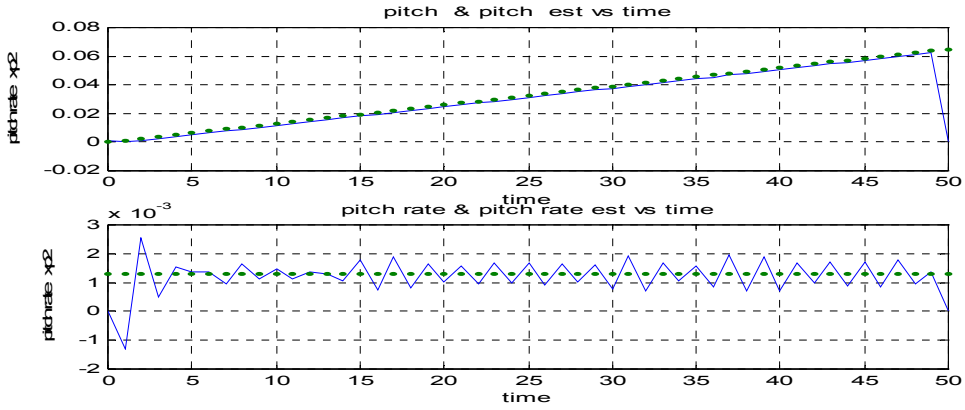


Figure 19: Pitch (q=0.01).

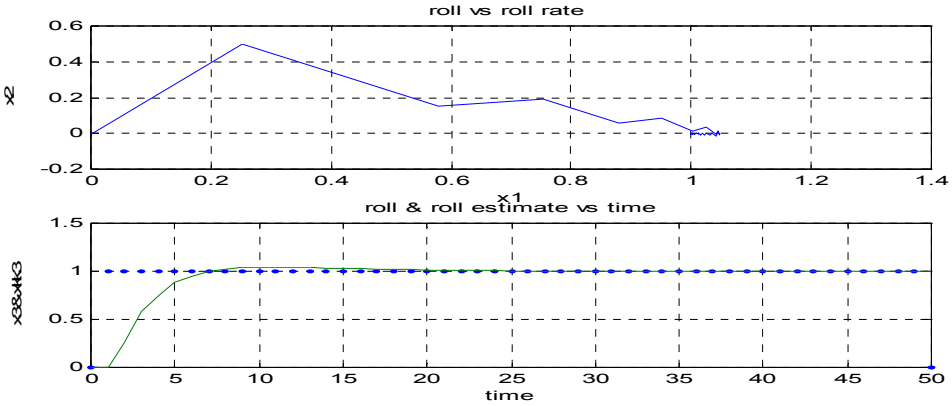


Figure 20: Roll (q=0.01).

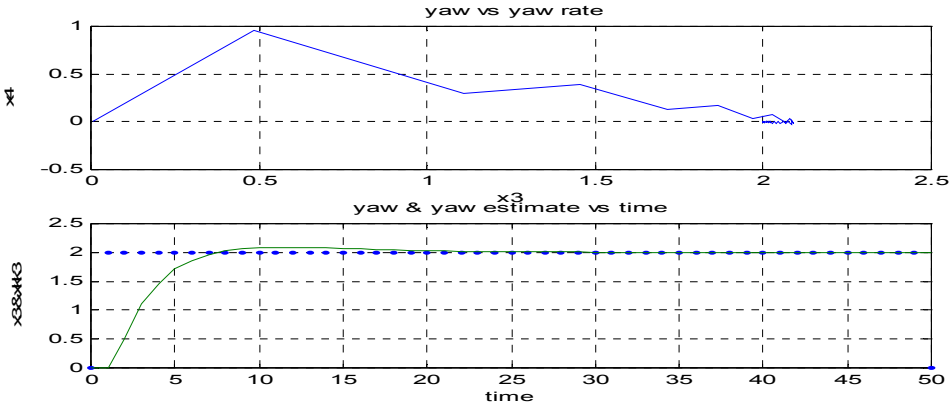


Figure 21: Yaw (q=0.01).

Figures (22), (23), and (24) represent constant gain Kalman filter with q=10.0.

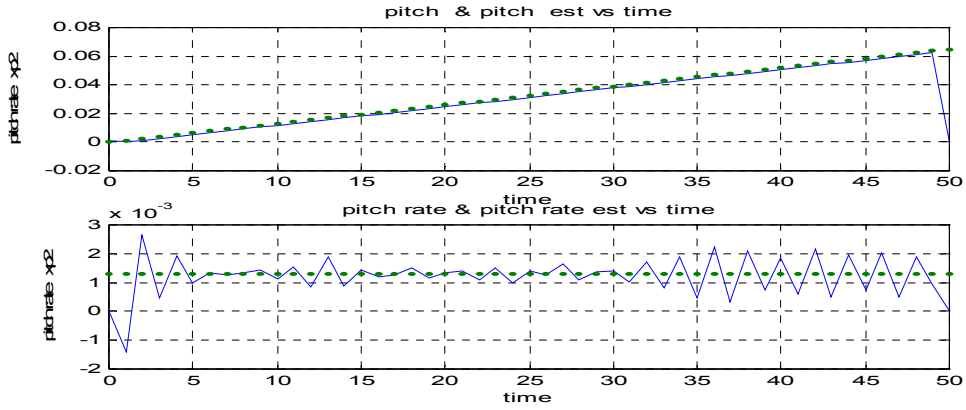


Figure 22: Pitch (q=10.0).

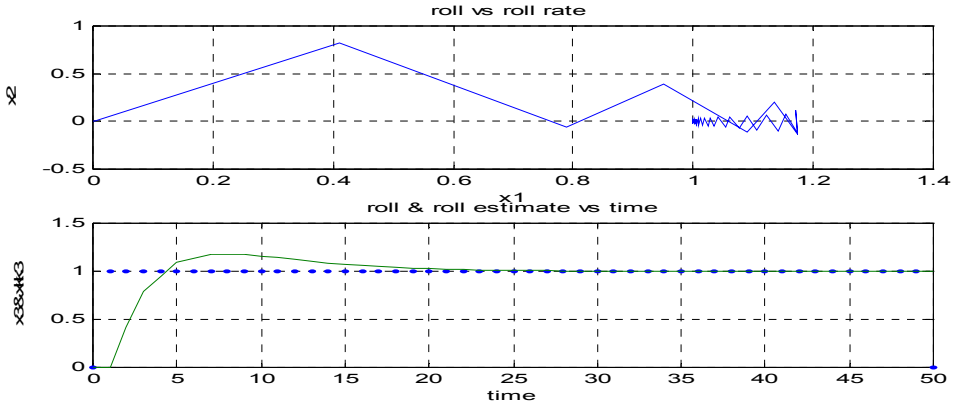


Figure 23: Roll (q=10.0).

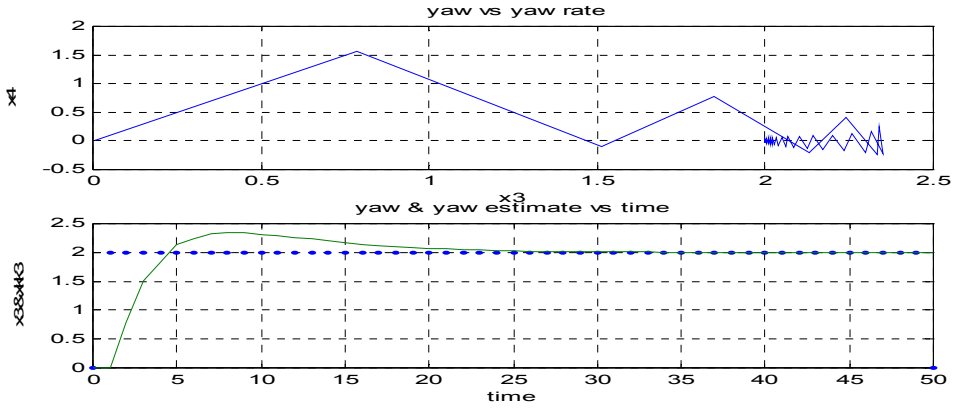


Figure 24: Yaw (q=10.0).

Figures (25), (26), and (27) represent constant gain Kalman filter with q=1000.0.

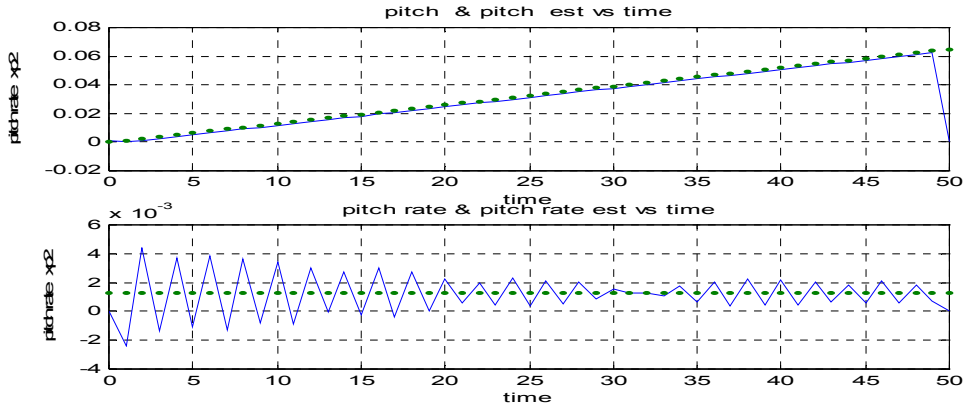


Figure 25: Pitch (q=1000.0).

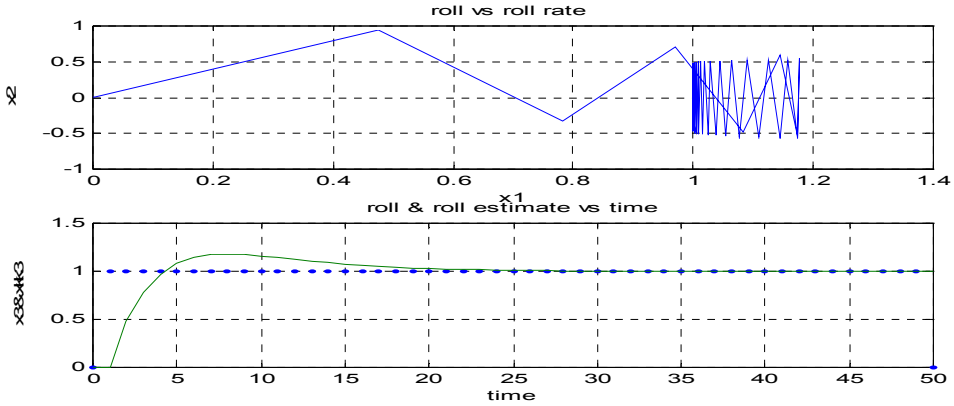


Figure 26: Roll (q=1000.0).

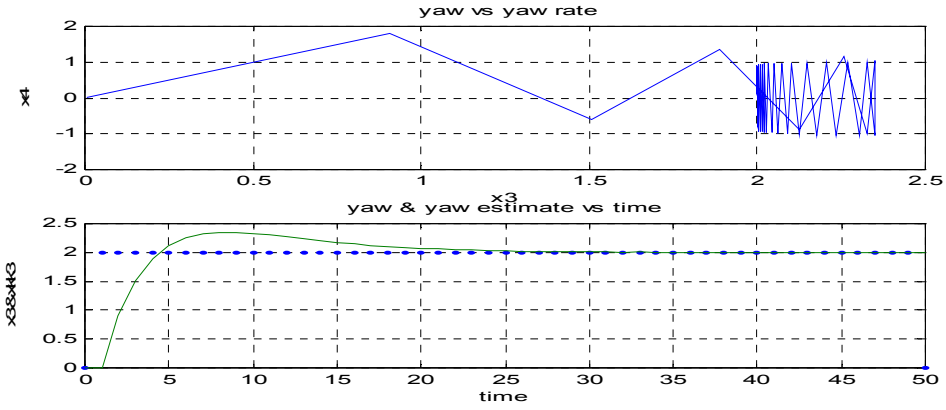


Figure 27: Yaw (q=1000.0).

THIS PAGE INTENTIONALLY LEFT BLANK

V. SUMMARY AND CONCLUSION

The process of the design and simulation of a small satellite is an essential part of technological development required to explore and test new design techniques and procedures. When exploring new technologies, it is essential to utilize available and proven current technologies, and test these technologies as backup systems to demonstrate design feasibility.

Additionally, the design and development of a optimal Kalman filter rate estimator to perform the essential requirement for attitude control determination can be useful for future development of a more complex rate estimator necessary to implement more advanced and complex control systems.

A. NPSAT-1 SUMMARY

The results obtained in this thesis are quite extraordinary. The controller uses a magnetic torque actuator to create the required torques.

The linear principle of superposition also allowed the removal of wheel speed changing, creating a nice constant speed wheel. The system was well behaved. The orbit inclination is also a concern (this approach will probably have problems with equatorial or polar orbits). There is an important detail that must be mentioned: the controller dead zone.

B. KALMAN FILTER RATE ESTIMATOR

This thesis has shown that a properly designed optimal rate estimator Kalman filter is effective and able to estimate body rate from a single star sensor. In addition,

these initial results prove that a single sensor couple with a proper rate estimator design can be used as backup or even primary attitude determination process.

For NPSAT-1, a single star sensor estimator will be an addition to the control system. This approach will be necessary, especially during the initial launch from the Pegasus, where after initial launch the spacecraft will tumbling at some pitch, roll, and yaw rates. The magnetometer and the magnetic torquers control, but would require, an additional vector which the star sensor can provide

C. FUTURE RESEARCH AREAS

The future of small satellite design and development is an interesting area. One approach would be to implement a MIMO controller using the state space approach. This could allow for more sophisticated techniques and possibly lower the design requirements of the sensors and actuators.

A second approach might be to design a controller that would eliminate the momentum wheel, replacing it with only three-axis magnetic torquers. This advance requires extensive investigation and simulation. However, initial testing indicates that this may be an adequate design approach for some future mission with less stringent pointing accuracy. Additional simulation would be necessary using a Monte Carlo approach for different αs and μs as required (a Monte Carlo approach is considered suitable).

Thirdly, new research on the development of an optimal estimator that includes 'sensor fusion.' Sensor fusion would combine all rates from every sensor and merge the data into a more accurate estimator. Due to the failure of aging satellites, design of a sensor fusion type of rate estimator could be useful to replace in-orbit, failing rate gyros,

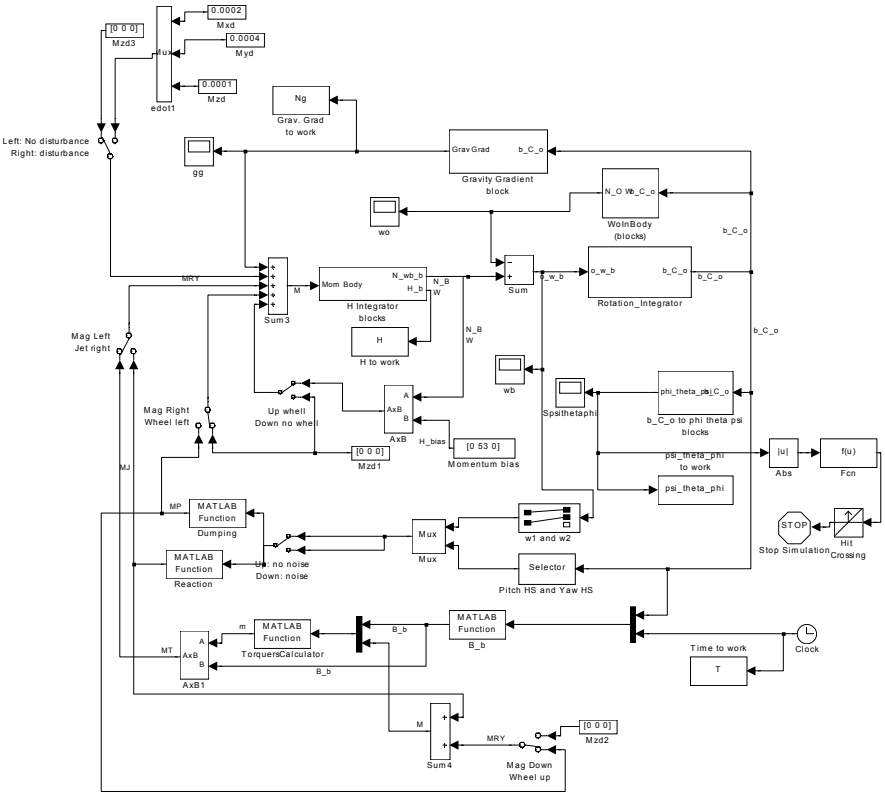
and allow for updating software packages that perform the rate estimation for gyros. Data streams would emanate from several different types of onboard sensor devices such as: star trackers, horizon sensors, magnetometers, and sun sensors

A fourth area of research would involve the concept of reverse time smoothing. Typically, the Kalman Filter uses only past and present observations, and is therefore a causal filter. This is ideal for real time systems such as satellites. However, for improved estimates, the additional computing power of modern satellites could be used to post-process old data. The smoothed past estimates could then be used in the Kalman Filter.

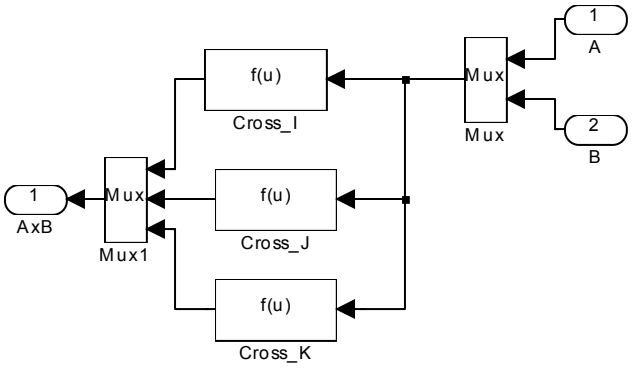
When considering a fixed interval smoother, several methods have been developed to post-process data. Three of the most common are 1) forward-backward smoother, 2) two-point boundary value approach, and 3) the Rauch-Tung-Streibel smoother. Additional work in this area could potentially offer some accuracy improvements.

THIS PAGE INTENTIONALLY LEFT BLANK

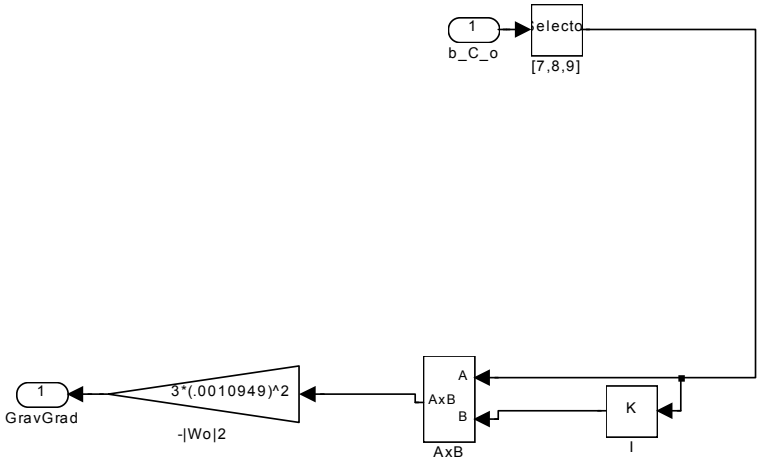
APPENDIX A: SIMULINK AND MATLAB CODE



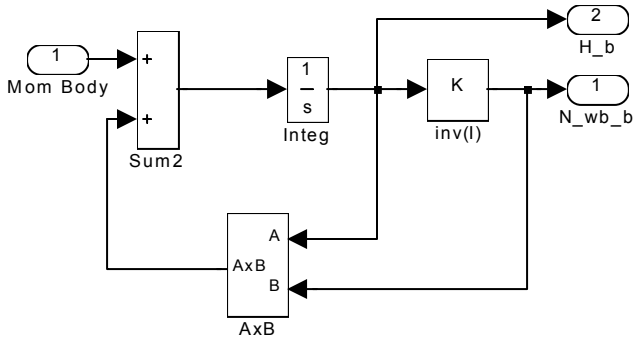
Appendix Figure 1: Simulink Block Diagram.



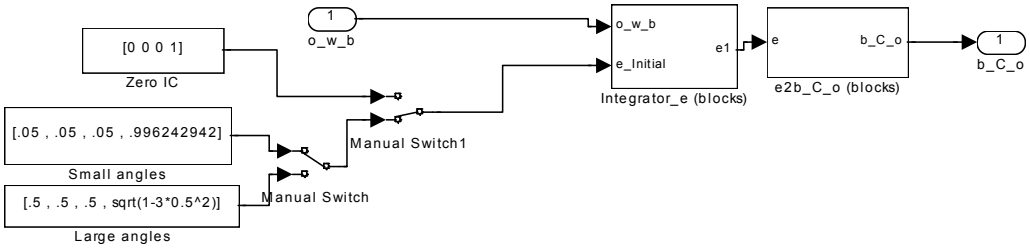
Appendix Figure 2: Cross Product.



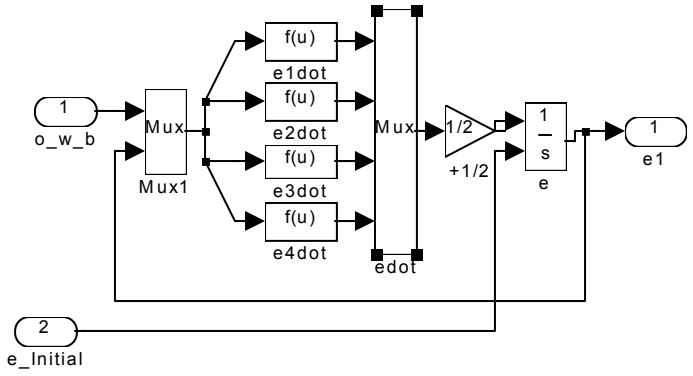
Appendix Figure 3: Gravity Gradient Block.



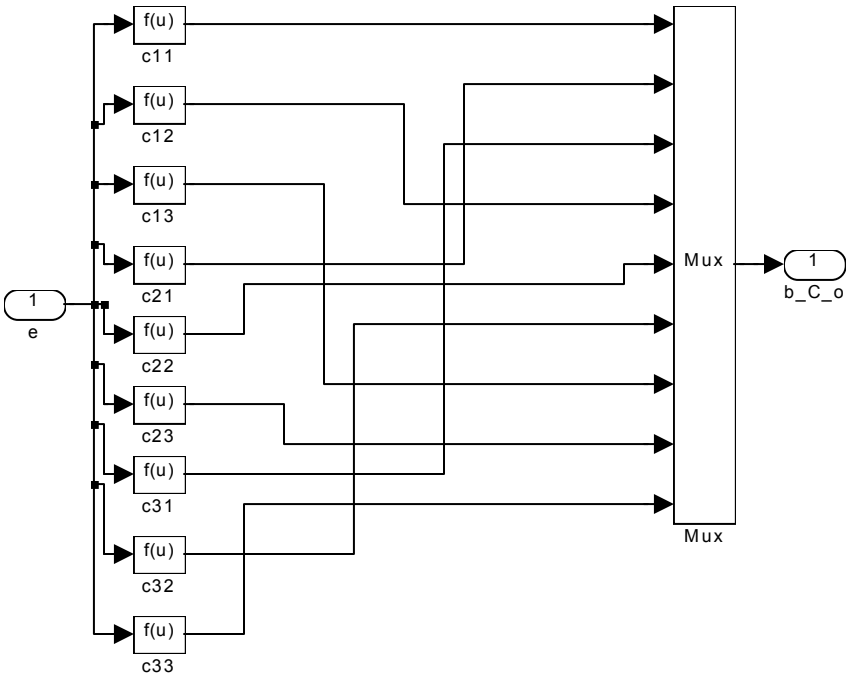
Appendix Figure 4: H-Block Integrator.



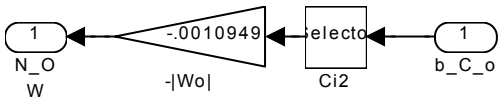
Appendix Figure 5: Rotation Integrator.



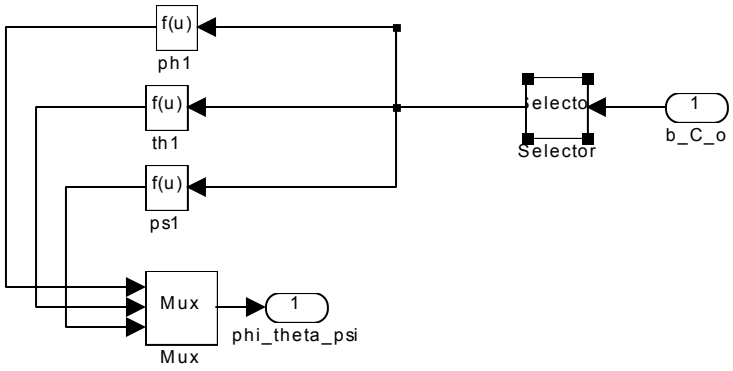
Appendix Figure 6: Rotation Integrator E-Block.



Appendix Figure 7: Rotation Integrator e2b_C_o.



Appendix Figure 8: w_n_body.



Appendix Figure 9: b_C_o to phi theta psi.

```
function M=Dumping(u)
%w1=u(1);
w2=u(2);
c13=u(3);
%c23=u(4);
K1=14;
K2=0.1146;
K3=1;
T=K2*(-K1*w2+c13);
M=K3*T*[0 1 0]';

function b_C_o=e2b_C_o(e)
e=reshape(e,4,1);
C=zeros(3,3);
C(1,1)=e(1)*e(1)-e(2)*e(2)-e(3)*e(3)+e(4)*e(4);
C(2,1)=2*(e(1)*e(2)-e(3)*e(4));
C(3,1)=2*(e(1)*e(3)+e(2)*e(4));
C(1,2)=2*(e(1)*e(2)+e(3)*e(4));
C(2,2)=e(2)*e(2)-e(1)*e(1)-e(3)*e(3)+e(4)*e(4);
C(3,2)=2*(e(2)*e(3)-e(1)*e(4));
C(1,3)=2*(e(1)*e(3)-e(2)*e(4));
C(2,3)=2*(e(2)*e(3)+e(1)*e(4));
C(3,3)=e(3)*e(3)-e(1)*e(1)-e(2)*e(2)+e(4)*e(4);
Error=0;
tol=1e-6;
if (abs(norm(C(1,:))-1)>tol)
   Error=1;
elseif (abs(norm(C(2,:))-1)>tol)
   Error=1;
elseif (abs(norm(C(3,:))-1)>tol)
   Error=1;
elseif (abs(norm(C(:,1))-1)>tol)
   Error=1;
elseif (abs(norm(C(:,2))-1)>tol)
   Error=1;
elseif (abs(norm(C(:,3))-1)>tol)
   Error=1;
end;
if (Error==1)
   disp('Error on b_C_o!!!Reduce time step or tighten
      tolerance.');
end;
b_C_o=reshape(C,9,1);

function M=GravityGrad(u)
b_C_o=reshape(u,3,3);
C=b_C_o(:,3);
I=[22.63 0 0; 0 24.67 0;0 0 11];
%I=[2246.87 0 0; 0 3202.94 0; 0 0 969.121];
wo=.0010949;
M=3*wo^2*cross(C,I*C);

function B=MagneticField(U)
b_C_o=U(1:9);
t=U(10);
alpha0=0;
```

```
u0=0;i=75*pi/180;
e=11*pi/180;
K=7.943e15;
w0=.0010949;
we=2*pi/(24*3600);
C=reshape(b_C_o,3,3);
alpha=alpha0+w0*t;
u=u0+we*t;
R=(6378.4+550)*1000;
r_o=[0 0 -1]';
Ca=cos(alpha);Sa=sin(alpha);
Cu=cos(u);Su=sin(u);
Ce=cos(e);Se=sin(e);
Ci=cos(i);Si=sin(i);
Cx=-Ca*(Ce*Si-Se*Ci*Cu)+Sa*Se*Su;
Cy= Ce*Ci+Se*Si*Cu;
Cz= Se*(Ca*Su-Sa*Cu*Ci)+Ce*Sa*Si;
M_o=[Cx,Cy,Cz]';
B_o=(K/R^3)*(3*(M_o'*r_o)*r_o-M_o);
B_b=C*B_o;

function M=ReactionJets(u)
w1=u(1);
%w2=u(2);
%c13=u(3);
c23=u(4);
K1= 800;
K2= 0.0573;
K3= 0.268;
K4=1;
Mx=-K2*(800*w1+c23);
Mz=-K3*Mx;
M=[Mx,0,Mz]';

function edot=we2edot(u)
w=u(1:3);
e=u(4:7);
w=reshape(w,3,1);
e=reshape(e,4,1);
if (abs(norm(e)-1)>1e-6)
   disp('Sum of squares of e'' not summing to 1!!');
end;
M=     [   0       w(3) -w(2) w(1);
        -w(3)        0   w(1) w(2);
         w(2)     -w(1)    0  w(3);
        -w(1)     -w(2) -w(3)   0];

edot=0.5*M*e;

function wo=WoInBody(b_C_o)
C=reshape(b_C_o,3,3);
wo=.0010949;
wo=-wo*C(:,2);

function wo=WoInBody(b_C_o)
```

```
C=reshape(b_C_o,3,3);
wo=.0010949;
wo=-wo*C(:,2);


%  Sat att rate est..
% x1 is roll,x2 is roll rate,  x3 is yaw, x4 is yaw rate,
      xp is pitch

%  orbit by discrete. This becomes the pitch observable
      xo(3,i)
Ao=[0 1 0 0;0 0 0 0;0 0 0 1;0 0 0 0]
Bo=[ 0 0;1 0;0 0;0 1]
Co=[1 0 0  0];
%N=-1;Tf=29400;dt=2;kmax=Tf/dt +1;
N=-1;Tf=38900;dt=2;kmax=Tf/dt +1;
[Phio,Delo]=c2d(Ao,Bo,dt)
uo=zeros(2,kmax);ho=zeros(1,kmax);xo=zeros(4,kmax);
yo=zeros(1,kmax);time=zeros(1,kmax);xcart=zeros(2,kmax);
xo(:,1)=[7439*0.62 0 0 0.00130]';
for (i=1:kmax-1)
    uo(:,i)=[xo(1,i)*xo(4,i)^2-
      (32.16/5280)*4000^2/(xo(1,i))^2;
-(2*xo(2,i)*xo(4,i))/(xo(1,i))];
    xo(:,i+1) = Phio*xo(:,i) +Delo*uo(:,i);
   time(i+1)= time(i) + dt;
end;
for (i=1:kmax)
  xcart(1,i)=xo(1,i)*sin(xo(3,i));
  xcart(2,i)=xo(1,i)*cos(xo(3,i));
 end
 w=0.00013;
 A=[0 1 0 0;0 0 0 -w;0 0 0 1;0 w 0 0];
B=[0 0; 1 0;0 0;0 1];
Bp=[0 1]';
dt=2;
pz=[0.78 0.79 0.77  0.76]';
[Phi,Del]=c2d(A,B,dt)
k=place(Phi,Del,pz)
ppp=eig(Phi)
pppp=eig(Phi-Del*k)
pause
[Phi,Del]=c2d(A,B,dt)
Ap=[0 1; 0 0]
[Phip,Delp]=c2d(Ap,Bp,dt)
p=[.85 .856]'
kd=place(Phip,Delp,p)
pp=eig(Phip+Delp*kd)
%pause
u=zeros(2,kmax);
up=zeros(1,kmax);
x=zeros(4,kmax);
xp=zeros(2,kmax);
y=zeros(1,kmax);
xf=zeros(4,kmax);
time=zeros(1,kmax);
```

```
zkkm1=zeros(2,1);
       % Random number generator
       v=randn(1,v);
zkkm1p=0;
P=50*eye(4);
Q=[(dt^4)/4  (dt^3)/2 0 0;(dt^3)/2 dt^2 0 0; ...
0 0 (dt^4)/4  (dt^3)/2;0 0 (dt^3)/2 dt^2]*0.0001;
R=[.0002 0;0 .0002];
H=[1 0 0 0;0 0 1 0];
Pp=50*eye(2);
Qp=[(dt^4)/4  (dt^3)/2;(dt^3)/2 dt^2]*0.0001
Rp=[.00002];
Hp=[1 0 ];
x(:,1)=[0.001; 0.002 ;0.003; 0.004];
xkk=zeros(4,kmax);
xkkm1=zeros(4,kmax);
xkkm1(:,1)=[0.0;0.0;0;0.0013];
xp(2,1)=0.0013;
xkkp=zeros(2,kmax);
xkkm1p=zeros(2,kmax);
xkkm1p(:,1)=[0;0];
zkkm1(:,1)=[0;0];
z=[x(1,1);x(3,1)];
for (i=1:kmax-1)
%  plant
   u(1,i)= -k(1,:)*x(:,i);
   u(2,i)= -k(2,:)*x(:,i);
   x(:,i+1) = Phi*x(:,i) + Del*u(:,i)  ;
   time(i+1)= time(i) + dt;
     % Random number generator
                v=randn(1,v);
                y(1,i+1)=0.000001*randn(1);
%  Kalman Filter
   G=P*H'*inv(H*P*H'+R);
   Pk=(eye(4)-G*H)*P;
   P=Phi*Pk*Phi'+Q;
   xkk(:,i)=xkkm1(:,i)+G*(z-zkkm1);
   xkkm1(:,i+1)=Phi*xkk(:,i);
   zkkm1=[xkkm1(1,i+1);xkkm1(3,i+1)];
   z=[x(1,i+1);x(3,i+1)];
   % from Kepler  H = mr1^2*w1 = mr2^2*w2....r2 = xo(2,i) =
      (r1^2*w1/w2)^(0.5)
   %r2=((7439*0.62)^2*0.0013)^0.5;
   Gp=Pp*Hp'*inv(Hp*Pp*Hp'+Rp);
   Pkp=(eye(2)-Gp*Hp)*Pp;
   Pp=Phip*Pkp*Phip'+Qp;
   zp=[xo(3,i)+y(1,i)];
   xkkp(:,i)=xkkm1p(:,i)+Gp*(zp-zkkm1p);

   up(i)= -(2*xo(2,i)*xkkp(2,i))/(xo(1,i));
   xp(:,i+1) = Phip*xp(:,i)  +Delp*up(i);
   xkkm1p(:,i+1)=Phip*xkkp(:,i)+ Delp*up(i);
   zkkm1p=[xkkm1p(1,i+1)];
end
clf
figure(1)
plot(time(1,:),y(1,:))
```

```
%pause
%clf
figure(2)
%subplot(221),
plot(x(1,:),x(2,:))
title('roll vs roll rate ')
xlabel('x1'), ylabel('x2'),grid
%subplot(222),
figure(3)
plot(time(1,:),xkk(3,:),'o',time(1,:),x(3,:))
title('yaw est vs time')
%subplot(223),
figure(4)
plot(x(3,:),x(4,:))
title(' yaw vs yaw rate')
xlabel('x3'), ylabel('x4'),grid
%subplot(224 ),
figure(5)
plot(time(1,:),xkk(4,:))
title('yaw rate est vs time ')
xlabel('xkk3'), ylabel('xkk4'),grid
%pause
%clf;
figure(6)
%subplot(221),
plot(time(1,:),xkk(1,:),time(1,:),x(1,:),'.');
xlabel('time'),ylabel('xkk1  &   x1');
title(['roll']);
%subplot(222),
figure(7)
plot(time(1,:),xkkp(2,:),' x',time(1,:),xp(2,:),'.')
xlabel('time'),ylabel('pitch rate   xp2');
title(['']);
grid;
%subplot(223),
figure(8)
plot(time(1,:),xkk(2,:),time(1,:),x(2 ,:),'.')
xlabel('time'),ylabel('roll rate    xkk2');
%subplot(224),
figure(9)
plot(time(1,:),xkkp(2,:))
xlabel('time'),ylabel('pitch rate est   xkkp2');
grid;
%pause
xxp=zeros(2,10);
xxkkp=zeros(2,10);
ttime=zeros(1,10);
for (i=1:70)
xxkk(2,i)=xkk(2,i);
xx(2,i)=x(2,i);
ttime(1,i)=time(1,i);
end
%clg
figure(10)
plot(time(1,:),xp(2,:),' x',time(1,:),xo(4,:),'o')
xlabel('time'),ylabel('pitch rate xo2 &  xp2');
title(['']);
```

```
%pause
figure(11)
plot(xcart(1,:),xcart(2,:))
title('orbit ');xlabel('x'), ylabel('y'),grid;
%pause
figure(12)
plot(ttime(1,:),xxkk(2,:),' x',ttime(1,:),xx(2,:),'o')
xlabel('time'),ylabel('roll rate   x2');
title(['']);
%pause
```

```
%MATLAB CODE
%Primary Code
%  Code written by Professor Hal Titus
%  Modified by LCDR John Vitalich 01 JUN 02
%  orbit by discrete. This becomes the pitch observable xo(3,i)
%  from Kepler  H = mr1^2*w1 = mr2^2*w2....r2 = xo(2,i) =
(r1^2*w1/w2)^(0.5)
%  r2=((7439*0.62)^2*0.0013)^0.5;

% constants
mu = 3.986e5             % Gravitational coefficient (km3/sec2)
w=0.00013;               % orbital frequency (rads/sec)
Tf=29400*2;             % seconds in one 12 hour pass
Tf=50;                       % seconds in one 12 hour pass
dt=1;
kmax=Tf/dt +1;
trackerr=6*pi/(3600*180);

% Kalman Filter Covariances and observation matrices
% roll yaw
P=1*eye(4);
q=0.01
Q=[(dt^4)/4  (dt^3)/2 0 0;(dt^3)/2 dt^2 0 0; ...
    0 0 (dt^4)/4  (dt^3)/2;0 0 (dt^3)/2 dt^2]*q;
R=[trackerr^2 0;0 trackerr^2];
H=[1 0 0 0;0 0 1 0];
% pitch
Pp=50*eye(2);
Qp=[(dt^4)/4  (dt^3)/2;(dt^3)/2 dt^2]*q;
Rp=[trackerr^2];
Hp=[1 0 ];

% initialize matrices
uo=zeros(2,kmax);ho=zeros(1,kmax);xo=zeros(4,kmax);
yo=zeros(1,kmax);time=zeros(1,kmax);xcart=zeros(2,kmax);
u=zeros(2,kmax);up=zeros(1,kmax);x=zeros(4,kmax);xp=zeros(2,kmax);
y=zeros(1,kmax);xf=zeros(4,kmax);time=zeros(1,kmax);zkkm1=zeros(2,1);
xkk=zeros(4,kmax);xkkm1=zeros(4,kmax);  z=zeros(2,1);
xkkp=zeros(2,kmax);xkkm1p=zeros(2,kmax);

% Define the Molniya orbit
Ao=[0 1 0 0;0 0 0 0;0 0 0 1;0 0 0 0];
Bo=[ 0 0;1 0;0 0;0 1];
Co=[1 0 0  0];
[Phio,Delo]=c2d(Ao,Bo,dt);

% xo1 is r (km), xo2 is r dot, xo3 is theta (rad), xo4 is theta dot
xo(:,1)=[7439 0 0 .0013]';
h=xo(1)^2*xo(4)                    % angular momentum
for (i=1:kmax-1)
    uo(:,i)=[xo(1,i)*xo(4,i)^2-mu/(xo(1,i))^2;
             -(2*xo(2,i)*xo(4,i))/(xo(1,i))];
    xo(:,i+1) = Phio*xo(:,i) +Delo*uo(:,i);
   time(i+1)= time(i) + dt;
end;
% convert orbit to rectangular co-ordinates for plotting
for (i=1:kmax)
```

```
  xcart(1,i)=xo(1,i)*sin(xo(3,i));
  xcart(2,i)=xo(1,i)*cos(xo(3,i));
  xc(1,i)= 6378*sin(xo(3,i));
  xc(2,i)= 6378*cos(xo(3,i));
end

% x1 is roll,x2 is roll rate,  x3 is yaw, x4 is yaw rate, xp is pitch
A=[0 1 0 0;0 0 0 -w;0 0 0 1;0 w 0 0];
B=[0 0; 1 0;0 0;0 1];
Ap=[0 1; 0 0];
Bp=[0 1]';

% place desired poles for roll and yaw
pz=[0.78 0.79 0.77  0.76]';
%pz=[.4 .41 .42 .43]';
% state transition matrix
[Phi,Del]=c2d(A,B,dt);

% gains and eigenvalues
k=place(Phi,Del,pz);
[klqr,s,elqr]=dlqr(Phi,Del,Q,R)         % discrete linear quadratic
regulator
%k=klqr;
ppp=eig(Phi);
pppp=eig(Phi-Del*k);

% place desired poles for pitch
p=[.85 .856]';
[Phip,Delp]=c2d(Ap,Bp,dt);

% gains and eigenvalues
kd=place(Phip,Delp,p);
[klqrp,s,elqr]=dlqr(Phip,Delp,Qp,Rp)         % discrete linear
quadratic regulator
%kd=klqrp;
pp=eig(Phip+Delp*kd);


% Kalman filter

% initial conditions
x(:,1)=[0.001; 0.002 ;0.003; 0.004];        % x is truth state
xkkm1(:,1)=[0.0;0.0;0;0.0013];              % z is truth measurement
%(includes meas. error)
zkkm1p=0;
xp(2,1)=0.0013;
xkkm1p(:,1)=[0;0];
zkkm1(:,1)=[0;0];
zp=.0013;

% run filter
for (i=1:kmax-1)
%  plant
   time(i+1)= time(i) + dt;


   % for roll and yaw
```

```
   G=P*H'*inv(H*P*H'+R);                              % Kalman Gain
   Pk=(eye(4)-G*H)*P;                                      % Covariance
Update
   P=Phi*Pk*Phi'+Q;                                        % Covariance
Prediction
   xkk(:,i)=xkkm1(:,i)+G*(z-zkkm1);                % State estimate update
   u(1,i)= k(1,:)*(xkk(:,i)-x(:,i));                       % Control Law
   u(2,i)= k(2,:)*(xkk(:,i)-x(:,i));                       % Control Law
   x(:,i+1) = Phi*x(:,i) + Del*u(:,i)  ;           % Update State
   xkkm1(:,i+1)=Phi*xkk(:,i)+ Del*u(:,i) ;         % Predict State
estimate

   roller=12*(randn(1)-.5)*pi/(3600*180);           % +/- 6 arcseconds of
random error
   yawer=12*(randn(1)-.5)*pi/(3600*180);           % +/- 6 arcseconds of
random error
   pitcher=12*(randn(1)-.5)*pi/(3600*180);          % +/- 6 arcseconds of
random error

   z=[1+roller;2+yawer];
   zkkm1=[xkkm1(1,i+1);xkkm1(3,i+1)];              % Update Measurement
estimate

   % for pitch
   Gp=Pp*Hp'*inv(Hp*Pp*Hp'+Rp);                     % Kalman Gain
   Pkp=(eye(2)-Gp*Hp)*Pp;                                  % Covariance
Update
   Pp=Phip*Pkp*Phip'+Qp;                                   % Covariance
Prediction
   xkkp(:,i)=xkkm1p(:,i)+Gp*(zp-zkkm1p);            % State estimate
update
   up(i)= -(2*xo(2,i)*xkkp(2,i))/(xo(1,i));                % Control Law
   xp(:,i+1) = Phip*xp(:,i)  +Delp*up(i);                  % Update State
   xkkm1p(:,i+1)=Phip*xkkp(:,i)+ Delp*up(i);        % Predict State
estimate
   zp=[xo(3,i)+pitcher];                                   % Update
Measurement
   zkkm1p=[xkkm1p(1,i+1)];                            % Update
Measurement estimate
end % Kalman loop


clf
figure(1)
plot(xcart(1,:),xcart(2,:),0,0,'*',xc(1,:),xc(2,:))
AXIS([-35000 35000 -60000 10000])
XLABEL('km'); YLABEL('km')

figure(2)
subplot(211),plot(x(1,:),x(2,:))
title('roll vs roll rate ')
xlabel('x1'), ylabel('x2'),grid

subplot(212),plot(time(1,:),xkk(1,:),'.',time(1,:),x(1,:))
title('roll & roll estimate vs time')
xlabel('time'), ylabel('x3 & xkk3'),grid
```

```
figure(3)
subplot(211),plot(x(3,:),x(4,:))
title(' yaw vs yaw rate')
xlabel('x3'), ylabel('x4'),grid

subplot(212),plot(time(1,:),xkk(3,:),'.',time(1,:),x(3,:))
title('yaw & yaw estimate vs time')
xlabel('time'), ylabel('x3 & xkk3'),grid

figure(4)
subplot(211)
plot(time(1,:),xkkp(1,:),time(1,:),xp(1,:),'.')
title(['pitch  & pitch  est vs time']);
xlabel('time'),ylabel('pitch rate   xp2');
grid;

subplot(212)
plot(time(1,:),xkkp(2,:),time(1,:),xp(2,:),'.')
title(['pitch rate & pitch rate est vs time']);
xlabel('time'),ylabel('pitch rate   xp2');
grid;

% Profile Generator
      %function xtrue=profile(kmax)
   %xtrue

% Random number generator

for i=1:kmax
y(1,i+1)=12*(randn(1)-.5)*pi/(3600*180);    % +/- 6 arcseconds of
random error
y(2,i+1)=12*(randn(1)-.5)*pi/(3600*180);    % +/- 6 arcseconds of
random error

xtrue(:,i)=[1;0;2;0;1+y(1,i+1); 2+y(2,i+1)];   % profile variable
end
```

LIST OF REFERENCES

1. *NPSAT-1 Preliminary Design Report*, Space Systems Engineering Students, Naval Postgraduate School, September 15, 1999.

2. Larson, W. J. and Wertz, J. R., *Space Mission Analysis and Design. Second Edition*, Microcosm, Inc., Torrance, CA and KIuwer Academic Publishers, Netherlands, 1992.

3. Brown, Robert R. and Hwang, Patrick Y. C., *Introduction to Random Signals and Applied Kalman Filtering 3rd ed.*, John Wiley & Sons, Inc., New York, NY, 1997.

4. Wiesel, William E., *Spaceflight Dynamics 2nd ed.*, McGraw-Hill, Inc., New York, NY, 1997.

5. Wertz, James R., *Spacecraft Attitude Determination and Control*, D. Reidel Publishing Company, Hingham, MA 1978.

6. Sidi, Marcel J., *Spacecraft Dynamics and Control*, Cambridge University Press, New York, NY, 1997.

7. Leonard, B. S., Class notes from AA 4871 Satellite Design Class, August 1994.

8. Leonard, B., *Spacecraft Design and Integration I (AA4870) Class Notes*, 1999.

9. Gelb, Arthur, *Applied Optimal Estimation,* MIT Press, Cambridge , MA, 1974.

10. Vogt, Jay D., " Attitude Determination of a Three Axis Stabilized Spacecraft Using Star Sensors", Thesis, Naval Postgraduate School, December 1999.

11. Agrawal, Brij N., *Design of Geosynchronous Spacecraft*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1986.

12. Brett, Daniel J., "Orbital Parameter Estimation Using an Extended Kalman Filter", Dissertation, Naval Postgraduate School, December 1992.

13. Fast Satellite. < http://www.sunland.gsfc.nasa.gov/smex/fast_top.html>, 1999.

14. Hablani, Hari B., "Pole-Placement Technique for Magnetic Momentum Removal of Earth-Pointing Spacecraft," *Journal of Guidance, Control, and Dynamics, Vol. 20, No.2,* pp. 268-275 March-April 1997.

15. Hashmall, J., Rowe, J., Sedlak, J., "Spacecraft Attitude Sensor Calibration From On-Orbit Experience," IEEE Proceeding, 1997.

16. Kaplan, Marshall H., *Modern Spacecraft Dynamics and Control*, John Wiley & Sons, Inc., New York, NY, 1976.

17. Kirk, Donald E., *Optimal Estimation: An Introduction to the Theory and Applications*, Class notes, 1975.

18. Larson, Wiley J. and Wertz, James R., *Space Mission Analysis and Design*, Microcosm Inc., Torrance, CA and Kluwer Academic Publishers, Netherlands, 1992.

19. Moon, Todd K. and Stirling, Wynn C., *Mathematical Methods and Algorithms for Signal Processing,* Prentice Hall, Upper Saddle River, NJ, 2000.

20. Ogata, Katsuhiko, *Modern Control Engineering 3rd ed.*, Prentice-Hall, Inc., Upper Saddle River, NJ, 1997.

21. Poker Flat Research Range. <http://www.pfrr.alaska.edu>, 1999.

22. Shorshi, G., Bar-Itzhack, I., "Satellite Autonomous Navigation and Orbit Determination Using Magnetometers," Proceeding of the 31st Conference on Decision and Control, Tucson, AZ., December 1992.

23. Solomon,S. C., .*The SNOE Spacecraft: Integration, Test, Launch, Operation, and On-orbit Performance,* 12th AIAA/USU Conference on Small Satellites, 1998.

24. Stickler, Craig A. and Alfriend, K.T., "An Elementary Magnetic Attitude Control System," *AIAA Journal of Spacecraft and Rockets, Vol. 13, No. 5,* pp.282-287 May 1975.

25. Student Nitric Oxide Explorer (SNOE). <http://lasp.colorado.edu/snoe, 1999.

26. Taranti, Christian, G. R., "A Computationally Efficient Algorithm for Disturbance Cancellation to Meet The Requirements for Optical Payloads in Satellites", Dissertation, Naval Postgraduate School, December 2001.

27. Travis, Henry D., "Attitude Determination Using Star Tracker Data with Kalman Filters", Thesis, Naval Postgraduate School, December 2001.

28. Wang, P., Shtessel, Y. B., Wang, Yong-Shian, " Satellite Attitude Control Using only Magnetorquers, " IEEE Proceedings, 1998.

29. Wolfe, Scott M., "Small Satellite Attitude Control for Sun-Oriented Operations Utilizing a Momentum Bias with Magnetic Actuators", Thesis, Naval Postgraduate School, March, 1995

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   8725 John J. Kingman Road, Suite 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library
   Naval Postgraduate School
   411 Dyer Road
   Monterey, CA 93943-5000

3. Department Chairman, Code EE
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5000

4. Professor Hal A. Titus
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5000

5. Professor Brij N. Agrawal, Code AA/Ag
   Department of Aeronautics and Astronautics
   Naval Postgraduate School
   Monterey, CA 93943-5000

6. SRDC Research Library, Code AA
   Department of Aeronautics and Astronautics
   Naval Postgraduate School
   Monterey, CA 93943-5000

7. John J. Vitalich
   186 Starfish Ct.
   Marina, Ca. 93933